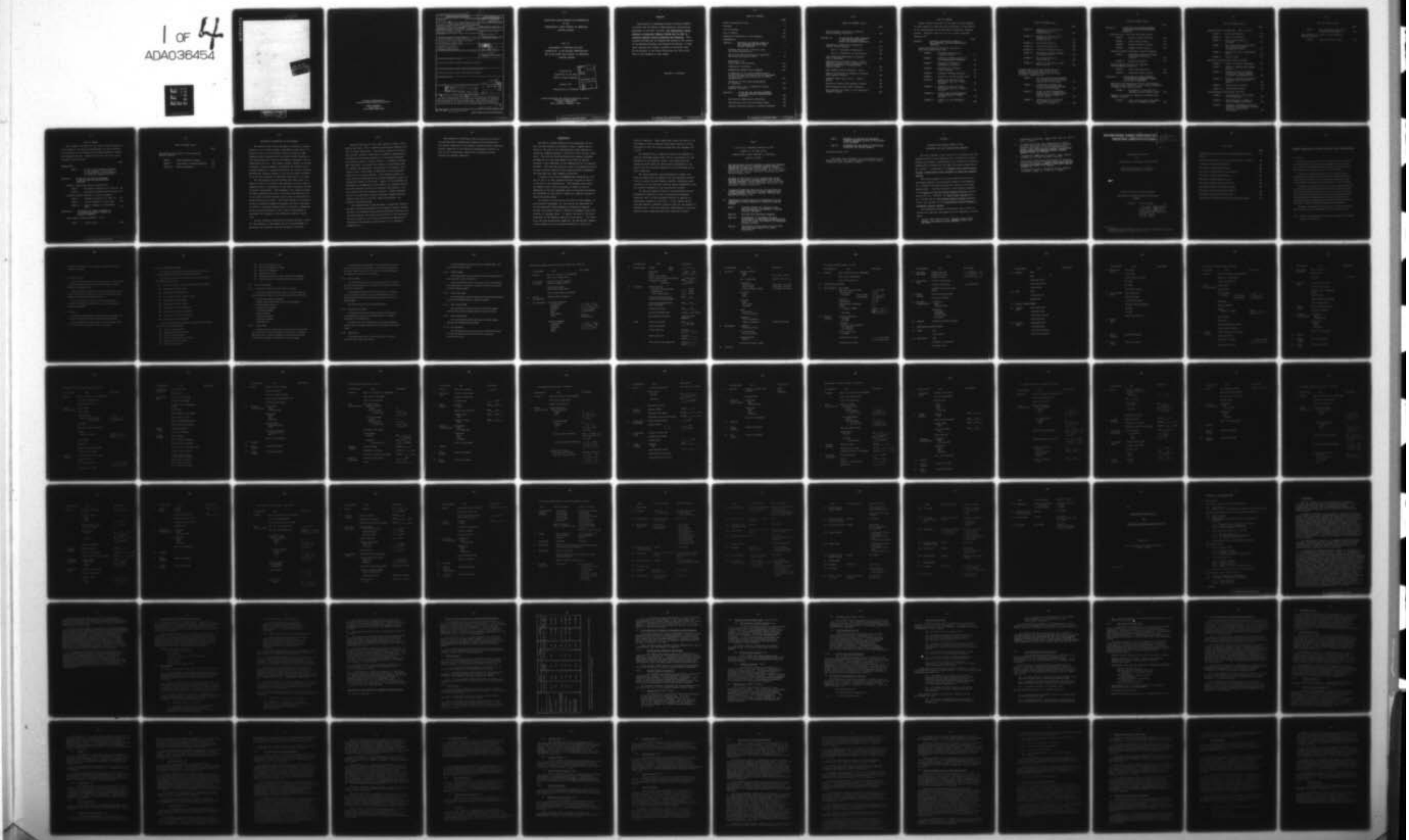
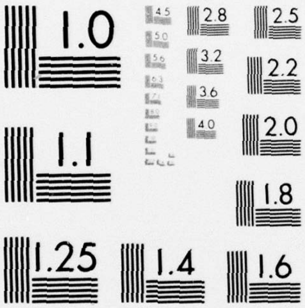
AD-A036 454 PURDUE UNIV LAFAYETTE IND PURDUE LAB FOR APPLIED IND--ETC F/G 9/2
SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIO--ETC(U)
JAN 77 N00014-76-C-0732

UNCLASSIFIED NL

1 OF 4
ADA036454




1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

(11)

This Report is Published as Part of
Engineering Experiment Station Bulletin 143 Series

Schools of Engineering
Purdue University
West Lafayette, Indiana 47907

11

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS
BEFORE COMPLETING FORM

1. REPORT NUMBER
NR049-388

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER
9 Final rept. 1 Feb 76-
31 Jan 77.

4. TITLE (and Subtitle) 6 SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIONAL PURDUE WORKSHOP ON COMPUTER SYSTEMS - PART III - DEVELOPMENT IN INTERFACES AND DATA TRANSMISSION, IN MAN-MACHINE COMMUNICATIONS AND IN THE SAFETY AND SECURITY OF INDUSTRIAL COMPUTER SYSTEMS.

5. TYPE OF REPORT & PERIOD COVERED
Final 2/1/76 - 1/31/77

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

8. CONTRACT OR GRANT NUMBER(s)
15 N00014-76-C-0732

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Purdue Laboratory for Applied Industrial Control
Schools of Engineering, Purdue University
West Lafayette, Indiana 47907

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Department of the Navy
Office of Naval Research
Arlington, Virginia 22217

12. REPORT DATE
11 Janu[redacted] [redacted]77

13. NUMBER OF PAGES
375 + xx

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)

15. SECURITY CLASS. (of this report)
12 372p.

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Approved for public release; distribution unlimited.

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This volume represents Part ~~III~~ 3 of a six volume set reproducing the major work accomplished by the International Purdue Workshop on Industrial Computer Systems during the past eight years. This material is reprinted from the Minutes of the several individual meetings of the Workshop and represents the work carried out by the standing committees of the Workshop.

408 244

SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIONAL PURDUE WORKSHOP ON INDUSTRIAL COMPUTER SYSTEMS

PART III

DEVELOPMENT IN INTERFACES AND DATA TRANSMISSION, IN MAN-MACHINE COMMUNICATIONS AND IN THE SAFETY AND SECURITY OF INDUSTRIAL COMPUTER SYSTEMS

Prepared for
Department of the Navy
Office of Naval Research

January 1977

Distribution is Unlimited

Purdue Laboratory for Applied Industrial Control
Schools of Engineering
Purdue University
West Lafayette, Indiana 47907

FOREWORD

This material is published as part of Contract N00014-76-C-0732 with the Office of Naval Research, United States Department of the Navy, entitled, *The International Purdue Workshop on Industrial Computer Systems and Its Work in Promoting Computer Control Guidelines and Standards*. This contract provides for an indexing and editing of the results of the Workshop Meetings, particularly the Minutes, to make their contents more readily available to potential users. We are grateful to the United States Navy for their great help to this Workshop in this regard.

Theodore J. Williams

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

LIST OF FIGURES

Figure numbers used here are the same as those assigned to these figures in their previous publication in the Minutes of the International Purdue Workshop on Industrial Computer Systems. Therefore they will not be in strict numerical sequence here.

Page

SECTION I - PROPOSALS AND WORKING PAPERS OF THE INTERFACES AND DATA TRANSMISSION COMMITTEE

Serial Line Sharing System for Industrial Real-Time Applications (SIR)

LIST OF FIGURES (Cont.)

A Comparison of Data Rate Capabilities of Various Interface Techniques Versus Requirement of Selected Processes and Levels of Control Implementation

LIST OF FIGURES (Cont.)

LIST OF FIGURES (Cont.)

Page

Computer Safety and Security - Back to Basics

Methods to Develop Safe Computer Systems

Safe Software by Functional Diversity

LIST OF FIGURES (Cont.)

The Guideline for Safety of the Industrial Computer Systems

LIST OF TABLES

Table numbers used here are the same as those assigned to these tables in the previous publications of these documents in the Minutes of the International Purdue Workshop on Industrial Computer Systems. Therefore they will not be in strict numerical succession here.

Page

INTRODUCTION

SECTION II - GUIDELINES AND RELATED DOCUMENTS OF THE MAN/MACHINE COMMUNICATIONS COMMITTEE

Committee Report Man-Machine Communication

Japanese Standard Operator's Console Guidebook

SECTION III - GUIDELINES AND OTHER DOCUMENTS OF THE SYSTEM RELIABILITY, SAFETY AND SECURITY COMMITTEE

Safe Computer Systems Hardware - Part 1

LIST OF TABLES (Cont.)

Page

The Guideline for Safety of the Industrial Computer Systems

BACKGROUND INFORMATION ON THE WORKSHOP

The *International Purdue Workshop on Industrial Computer Systems*, in its present format, came about as the result of a merger in 1973 of the Instrument Society of America (ISA) Computer Control Workshop with the former Purdue Workshop on the Standardization of Industrial Computer Languages, also co-sponsored by the ISA. This merger brought together the former workshops' separate emphases on hardware and software into a stronger emphasis on engineering methods for computer projects. Applications interest remains in the use of digital computers to aid in the operation of industrial processes of all types.

The ISA Computer Control Workshop had itself been a renaming in 1967 of the former Users Workshop on Direct Digital Computer Control, established in 1963 under Instrument Society of America sponsorship. This Workshop in its annual meetings had been responsible for much of the early coordination work in the field of direct digital control and its application to industrial process control. The Purdue Workshop on Standardization of Industrial Computer Languages had been established in 1969 on a semiannual meeting basis to satisfy a widespread desire and need expressed at that time for development of standards for languages in the industrial computer control area.

The new combined international workshop provides a forum for the exchange of experiences and for the development of guidelines and proposed standards throughout the world.

Regional meetings are held each spring in Europe, North America and Japan, with a combined international meeting each fall at Purdue University. The regional groups are divided into several technical committees to assemble implementation guidelines and standards proposals on specialized hardware and software topics of common interest. Attendees represent many industries, both users and vendors of industrial computer systems and components, universities and research institutions, with a wide range of experience in the industrial application of digital systems. Each workshop meeting features tutorial presentations on systems engineering topics by recognized leaders in the field. Results of the workshop are published in the Minutes of each meeting, in technical papers and trade magazine articles by workshop participants, or as more formal books and proposed standards. Formal standardization is accomplished through recognized standards-issuing organizations such as the ISA, trade associations, and national standards bodies.

The International Purdue Workshop on Industrial Computer Systems is jointly sponsored by the Automatic Control Systems Division, the Chemical and Petroleum Industries Division, and the Data Handling and Computations Division of the Instrument Society of America, and by the International Federation for Information Processing as Working Group 5.4 of Technical Committee TC-5.

The Workshop is affiliated with the Institute of Electrical and Electronic Engineering through the Data Acquisition and Control Committee of the Computer Society and the Industrial Control Committee of the Industrial Applications Society, as well as the International Federation of Automatic Control through its Computer Committee.

INTRODUCTION

The Office of Naval Research of the Department of the Navy has made possible an extensive report, summary and indexing of the work of the International Purdue Workshop on Industrial Computer Systems as carried out over the past eight years. The work has involved twenty-five separate workshop meetings plus a very large number (over 100) of separate meetings of the committees of the workshop and of its regional branches. This work has produced a mass of documentation which has been severely edited for the original minutes themselves and then again for these summary collections.

A listing of all of the documentation developed as a result of the U. S. Navy sponsored project is given in Table I at the end of this Introduction. The workshop participants are hopeful that it will be helpful to others as well as themselves in the very important work of developing guidelines and standards for the field of industrial computer systems in their many applications.

In contrast to the previous two Parts of this Summary, or more correctly anthology, of the work of the Committees of the International Purdue Workshop on Industrial Computer Systems, the present volume is devoted to hardware rather than software or language items. It reports the work of the three committees of the Workshop devoted to such topics: the Interfaces and Data Transmission Committee, the Man-Machine Communications Committee and the Systems Reliability, Safety and

Security Committee. These committees formed the basis of the ISA Computer Control Workshop which began meeting at Purdue University in May 1972 and was merged with the language work in 1973.

The Workshop has no committee studying the subjects of computer mainframe design since this is considered to be the perogative of the computer vendor. Any design would be acceptable which meets the operational requirements of the process and the interface standards to be established by the above committees.

The third committee, System Reliability, Safety and Security Committee is considering the very important problem of how to assure the very highest possible availability and operability of an industrial computer system commensurate with the required economics of the installation involved.

The American Regional Branch of the Interfaces and Data Transmission Committee is also constituted as Standards Committee, SP72, of the Instrument Society of America for developing standards in this area. It also serves as the cognizant American technical advisory group for the ISO/TC 97/SC13/WG1 work in this area entitled, "Description of Interface Between Process Computing System and Technical Process".

TABLE I

A LIST OF ALL DOCUMENTS PRODUCED IN THIS SUMMARY OF THE WORK OF THE INTERNATIONAL PURDUE WORKSHOP ON INDUSTRIAL COMPUTER SYSTEMS

1. *The International Purdue Workshop on Industrial Computer Systems and Its Work in Promoting Computer Control Guidelines and Standards*, *Report Number 77*, Purdue Laboratory for Applied Industrial Control, Purdue University, West Lafayette, Indiana, Originally Published May 1976, Revised November 1976.

2. *An Index to the Minutes of the International Purdue Workshop on Industrial Computer Systems and Its Predecessor Workshops*, *Report Number 88*, Purdue Laboratory for Applied Industrial Control, Purdue University, West Lafayette, Indiana, January 1977.

3. *A Language Comparison Developed by the Long Term Procedural Languages Committee - Europe*, *Committee TC-3 of Purdue Europe*, Originally Published January 1976, Republished October 1976.

4-9. *Significant Accomplishments and Documentation of the International Purdue Workshop on Industrial Computer Systems*.

Part I - *Extended FORTRAN for Industrial Real-Time Applications and Studies in Problem Oriented Languages*.

Part II - *The Long Term Procedural Language*.

Part III - *Developments in Interfaces and Data Transmission*, *in Man-Machine Communications and in the Safety and Security of Industrial Computer Systems*.

Part IV - *Some Reports on the State of the Art and Functional Requirements for Future Applications*.

Part V - Documents on Existing and Presently Proposed Languages Related to the Studies of the Workshop.

Part VI - Guidelines for the Design of Man/Machine Interfaces for Process Control.

All dated January 1977.

The latter seven documents are also published by the Purdue Laboratory for Applied Industrial Control, Purdue University, West Lafayette, Indiana.

SECTION I

PROPOSALS AND WORKING PAPERS OF THE INTERFACES AND DATA TRANSMISSION COMMITTEE

The first document in this section is a proposal from the Japanese Branch of this Committee for a standard way of documenting the technical specification for a particular industrial application in relation to the input and output connections to the process. It appeared in the Minutes of the Second Annual Meeting, International Purdue Workshop on Industrial Computer Systems.

The second document is the most recent version of the developing proposal of the European Branch of the Committee for a "Serial Line Sharing System for Industrial Real-Time Applications". Previous versions of this proposal appeared in the Minutes of the 1975 Spring Regional Meetings (Attachment E-CI-B, pp. 138-146); the Third Annual Meeting (Part I, pp. 223-250) and the 1976 Spring Regional Meeting (Appendix E-IV-C, pp. 187-222) of the International Purdue Workshop on Industrial Computer Systems.

The remaining documents included here are a series of smaller but important developments of the Committee as listed below:

1. "Onsite Remote Multiplexing", Minutes Second Purdue Meeting, ISA Computer Control Workshop, Insert V-2, pp. 63-65.

2. "Independent Interfaces", Ibid, Insert XII, pp. 105-111, by R. L. Curtis.

3. "A Comparison of Data Rate Capabilities of Various Interface Techniques versus Requirements of Selected Processes and Levels of Control Implementations", Minutes, 1974 Spring Regional Meeting, International Purdue Workshop on Industrial Computer Systems, Appendix III-IX, pp. 261-266.

4. "Implementing CAMAC Serial Highways", Ibid, Appendix III-VIII, pp. 251-260, by Dale W. Zobrist.

5. "Discussion of Functional Requirements of Interfaces and Data Transmission", Minutes Third Annual Meeting, International Purdue Workshop on Industrial Computer Systems, pp. 90-96, by T. Tohyama.

6. "A Comparative Look at Industrial Process Computer Interfaces", Ibid, pp. 97-106, by G. Merkel.

INTERNATIONAL PURDUE WORKSHOP ON INDUSTRIAL COMPUTER SYSTEMS

PURDUE LABORATORY FOR
APPLIED INDUSTRIAL CONTROL
102 Michael Golden
Purdue University
West Lafayette, Indiana 47907, USA
317/494 8425

Please reply to:

STANDARD DESCRIPTION
OF
PROCESS INPUT AND OUTPUT SPECIFICATION

(If there are any questions or comments on these documents, please let us know.)

Technical Committee on Process Interface

Japan Electronic Industry Development Association
(JEIDA)

Chairman: Takashi Tohyama

(Takashi Tohyama)

Instrumentation and Control Dept.
Chiyoda Chem. Eng. & Const. Co.,
No. 1580, Tsurumi,
Yokohama, JAPAN

Affiliations
Purdue University
Instrument Society of America through Data Handling and Computations, Chemical and Petroleum Industries, and Automatic Control Divisions
International Federation for Information Processing as Working Group, WG 5.4. Common and/or Standardized Hardware and Software Techniques of Technical Committee, TC 5, Computer Applications in Technology

List of Drafts

STANDARD DESCRIPTION OF PROCESS INPUT AND OUTPUT SPECIFICATION

Introduction

The technical committee on Process Interface which is organized by JEIDA (Japan Electronic Industry Development Association), has been formed and issued the standard description of Process Input and Output Specification as JEIDA report no., 49-A-82 (1974).

To make this standard specification, this committee made an investigation into Process I /O's specification of Japanese industrial computer system vendors. These results of investigation by using the questionnaire was published as JEIDA report no., 48-A-70 (1973).

Thereafter, the committee found that to make a standard specification of a process input and output interface was very different in each vender and to find out the requirements or necessity in the near future was very difficult according to the recent advance of interface communication procedures or systems (e.g. Data heighway system (line shearing procedure), Advanced solid state device including LSI microprocessor, Computer compartible instrumentation and so on.)

Therefore, the committee has worked out a standardization of the descriptive formats of the specification for Process I /O in 1973.

Note: In 1974, the committee are working to investigate a line shearing system for industrial use.

Standard Description of Process Input and Output Specification (Japanese Proposal)

1. Purpose and Usage

This standard contains the descriptive formats of the interface between the industrial process and the input/output devices of industrial computing system.

This is prepared to make easily the specifications by user and good communications in each other, and to accelerate the standardization of process input/output interface.

For easy utilization, the items of the specification are selected the functions and characteristics on the termination to industrial processes.

2. Reference

To define the characteristics of process input and output, the standard testing procedure and the definitions are required.

Herein, Hardware Testing of Digital Process Computer (ISA RP 55-1, 1971), Recommended Practice is referred.

3. Scope of Standard Description

The formats of the specification form are prescribed to use easy and defined according functional characteristics.

The process input and output interface units are specified in the following tenth forms.

(1) General description of Process Input and Output System.

(2) Description of Analog Input

(3) Description of Analog Output

(4) Description of Analog Control Output

(5) Description of Digital Input

(6) Description of Digital Output

(7) Description of Pulse Train Input

(8) Description of Pulse Train Output

(9) Description of Pulse Width Output

(10) Description of Interrupt Input

These Process I/O descriptions are constituted the following basic items except Analog Input.

(a) Purpose and application
(b) Input (Output) characteristics
(c) Electric Characteristics
(d) Dynamic Response Characteristics
(e) Operational Characteristics
(f) Safety Characteristics

(g) Structural Characteristics

(h) Special and Optional functions

(i) Basic Block Diagram

(j) Equivalence Circuit

(k) Backup Characteristics (Only Analog Control Output)

(l) Block Diagram for Procedure (Only Interrupt Input)

3.1. General Information

3.1.1. General Description of Process Input and Output.

General Description of Process Input and Output covers the common functions of process input and output devices as follows.

- Permissible system connection
- System configuration (Basic system block diagram)
- Environmental condition
- Power supply condition
- Grounding condition
- Cable condition
- Structural condition
- Installing condition
- Miscellaneous

3.1.2. Analog Input

The informations of analog input consist of the basic specification which include common interface for multiplexer, amplifier, ADC and control, and the individual specification which include termination and signal conditioning for each signal types.

The specification of amplifier is not described on account of unnecessary for user's view points. But, the specifications for accuracy, speed and noise are described as warrantable values including the characteristics of amplifier.

3.1.3. Analog Output

The informations of analog output are considered as characteristics and specifications on termination for external connection of process output unit.

The interfaces between CPU and Analog output termination are in existense many kinds of types and variations, so it is diffical to describe the characteristics of interface circuits for analog output.

It is defined to easy use by application's user.

3.1.4. Analog Control Output

The informations for DDC (Direct Digital Control) Output are considered as Analog Control Output which is arranged separately from analog output.

For DDC output, the informations of the backup capability and the speciale designed analog out units are added to the analog output.

3.1.5. Digital Input

This digital input covers the informations for electronic input and contact input of status bit.

The pulse input is not considered in this digital input. But it is defined as Pulse Input.

3.1.6. Digital Output

The digital output covers the informations for electronic output and contact output of status bit.

The pulse duration and/or train output are not considered in this digital output. But these are defined as Pulse Train Output or Pulse Duration Output.

3.1.7. Pulse Train Input

The informations of pulse train input are shown input interface for a series of pulses and/or a number of pulses.

3.1.8. Pulse Train Output

The informations of pulse train output are shown output interface for a series of pulses and/or a number of pulses.

3.1.9. Pulse Width Output

The informations of pulse width output are shown output interface for variable duration pulses.

3.1.10 Interrupt Input

The informations of interrupt input are defined the external interructive signal processing of the process requiring immediate attension.

General Description of Process Input and Output Unit (Form 1)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
2	Permissible Connection	Interface between computer (Channel, adaptor type etc)	
		Data handling made (Procedure of operation)	
		Direct and/or Remote connection	
3	System Configuration	(Shown by Block diagram)	
4	Environmental	*Operating Conditions:*	
		Temperature	______to______°C
		Humidity	______to______%RH
		Vibration	___ ___Hz(or___ G)
		Shock	____G____msec
		Dust	___mg/m^3(max.)
		Atmosphere	
		Misc.	Altitude, Radioactivity
		Storage Conditions	
		Temperature	______to______°C
		Humidity	______to______%RH
		Vibration	___ ___Hz(or___G)
		Shock	_____G_____msec
		Misc.	

	Classification	Item	Description
5	Power supply	Voltage (AC) (DC)	_____V ± ____V
		Frequency	____Hz I ____Hz
		Phase	____ϕ _____wires
		Type of Termination	
		Permissible power failure interval	____msec (max.)
		Backup power supply	YES ____ NO ____
		" distortion	_____% (max)
6	Grounding	Required ground	
		Safety (or Frame) ground	_____ (max)
		Signal ground	_____ (max)
		Power supply ground	_____ (max)
		Common grounding between instrument and signal ground	YES ____ No ____
		Common grounding between frame and signal	YES ____ No_____
		Shielding for frame	YES _____ No____
		Ground of shielded cable	P I/O____, Inst. Panel
		Block diagram of ground	(shown by Block diagram)
7	Cable	Conductor of Cable	Size___or_____ ______m (max)
		Shield and Isolation	
		Analog signal line	Twisted__________ Shield ___________
		Digital signal line	Twisted__________ Shield ___________
		Pulse and Interrupt signal line	Twisted__________ Shield ___________

	Classification	Item	Description
8	Structure	Standard structure	
		Type	
		Size	W_____xH____xD____
		Max. configuration	Max. No. of Units____
		Connecting	
		Analog signal	Terminals: Connector:
		Digital signal	Terminals: Connector:
		Pulse & Interrupt signal	Terminals: Connector:
		Terminals	
		Type	
		Size	
		Connector	
		Type	
		No. of Pins	
		Finished	
		Color	
		Color code	
		Finishing touches	
		Packaging	
		Type of Enclosure	
		Layout of Enclosure	(Shown by drawing)
9	Installation	Foundation (Weight and Base)	
		Air conditioning (Heat up amount)	
		Noise protection (Noise)	
		Recommended power supply	
10	Remarks		

Description of Analog Input (Form 2)

	Classification	Item		Description
1.	Purpose	Name of Computer to be applicable		
		Basic type of Application		
		Type of Terminal Unit		
2.	BASIC SPECIFICATION			
2. 1	Input Condition	Input range		
		Max. input points per ADC		_____Pts/ADC
		Total Accuracy		_____%
		Warranty	Temp. range	_____to_____°C
			Running Hour	_____Hr (max.)
		Linearity		______%
		Repeatability		______%
		Allowable Input Impedance		______
		Coding		
		Analog vs. Digital		Lower_____vs. ____
		"		Upper_____vs. _____
		With sign		YES_____No______
2. 2	Electric condition	Common mode error		
		DC CMR		
		AC CMR		
		Crosstalk		
		Common mode crosstalk		
		DC crosstalk		
		AC crosstalk		
		Normal mode error		
		AC NMR		
		Allowable Overvoltage		______Common Mode
				______Normal Mode
		Grounding at External		

	Classification	Item	Description
2. 3	Input Rate (Response)	Random scan rate	____Pts/sec(____ s)
		Sequental scan rate	____Pts/sec(____ s)
		Repeat scan rate	____Pts/sec(____ s)
2. 4	Operational Mode	Transfer control mode	
		Transfer block length	____Words/Block
		Transfer word configuration (code)	
2. 5	Safety function	Error checking functions	
		Protection functions	
2. 6	Structure (Configuration)	Connections	
		Type	
		Terminals size	
		Module unit	
		Terminals	
		Multiplexer	
		Unit (or Card)	
		Enclosure	
2. 7	Optional	Optional (or Special) functions	
3.	INDIVIDUAL SPECIFICATION		
3. 1	Filter	Type	
		Time constant	
3. 2	Multiplexer	Type	
		Multiplexer configuration	
		Multiplexer rate	

	Classification	Item	Description
3. 3	Amplifier	Type	
		Gain	
		Input signal level	
		Output signal level	
		Gain selection	
3. 4	ADC	Type	
		Conversion rate	
		Output code	
4.	SIGNAL CONDITIONING		
4. 1	Voltage Input	Input signal level	
		Conversion type	
		Conversion accuracy	
		Input impedance	
4. 2	Current Input	Input signal level	
		Conversion type	
		Conversion accuracy	
		Open circuit detection	

	Classification	Item	Description
4.3	Registance Input	Input range	
		Circuit type	
		Insulation between inputs	
		DC CMR	
		AC CMR	
		Conversion accuracy	
4.4	Thermocouple Input	Input signal level	
		Conversion type	
		Conversion accuracy	
		Thermocouple compensation	
		DC CMR	
		AC CMR	
		Open circuit detection	
4.5	Special Input		
5	Block Diagram	(shown by drawing)	
6	Input Circuit	(shown by drawing)	

Description of Analog Output (Form 3)

	Description	Item		Description
1	Purpose	Name of Computer to be applicable		
		Basic type of Application		
		Type of Terminal Unit		
2	Output Characteristics	Output range		
		Output rating		
		Total accuracy		______%
		Warranty	Temp. range	_____to_____°C
			Running Hour	_____Hr (max.)
		Resolution		
		Output impedance		
		Coding Digital vs. Analog		Lower_____vs. ____
		"		Upper_____vs. _____
		Initial status		
		Droop rate		
		Transient change per month		
		Power supply condition		
3	Electric condition	Common mode rejection ratio		
		Output noise level		
		Alowable overvoltage		______Common Mode
				______Normal Mode
		Grounding at External		

	Classification	Item	Description
4	Output rate (Response)	Max. output rate	
		Settling time	
		Slew rate	______ V/ Msec
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer word configuration (Code)	
		Output set-up	
		Output buffering and holding	
		Type of output circuit	
6	Safety function	Error checking functions	
		Protection functions	
7	Structure (Configuration)	Connections Type Terminals size	
		Module unit Card Unit Enclosure	
		Max. no of output point	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Output Circuit	(shown by drawing)	

Description of Analog Control Output (Form 4)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Output characteristics	Output range	
		Output rating	
		Total accuracy	______%
		Warranty Temp. range	_____to_____°C
		Running Hour	_____Hr (max.)
		Resolution	
		Maximum output change/Instr.	
		Coding	
		Digital vs. Analog	Lower_____vs. ____
		"	Upper_____vs. _____
		Initial status	
		Droop rate	
		Transient change per month	
		Power supply condition	
3	Electric Condition	Common mode rejection ratio	
		Output noise level	
		Allowable overvoltage	______Common Mode
			______Normal Mode
		Grounding at External	

	Classification	Item	Description
4	Output rate (Response)	Conversion time	
		Settling time	
5	Operational Mode	Transfer control mode	
		Transfer block length	
		Transfer word configuration (Code)	
		Output setup	
		Output buffering and holding	
		Type of output circuit	
		Output data feedback read in	
		Status information read in	
6	Safety function	Checking functions	
		Protection functions	
7	Backup function	Manual station	
		DPC to Manual switching	
		Manual to PDC switching	
		Manual output manupulation rate	
		Portable manual station	
		Backup control station (BUC: Backup control)	
		DOC to BUC switching	
		BUC to DOC switching	
		BUC to Manual switching	

	Classification	Item	Description
7	(Cont.)	Manual to BUC switching	
		Set point tracking	
		Process variable tracking	
		Manual output manupulation rate	
		Portable manual station	
8	Structure (Configuration)	Connections	
		Type	
		Terminals size	
		Module unit	
		Card	
		Unit	
		Enclosure	
		Manual station	
		Number per Unit	
		Size	
		Backup control station	
		Number per Unit	
		Size	
		Max. no of output point	
9	Optional		
10	Block Diagram	(shown by drawing)	
11	Output Circuit	(shown by drawing)	

Description of Digital Input (Form 5)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Input characteristics	Input signal level	
		High level (Make)	
		Logical	0 : 1
		Load (Contact)	__________
		" (Electronic)	__________ V
		" (")	__________mA
		Low level)Break)	
		Logical	0 : 1
		Load (Contact)	__________
		" (Electronic)	__________V
		" (")	__________mA
		External contact rating	
		Power source	
		Internal	DC_____V±____V
			_____mA/point
		External	DC_____V±____V
		(Internal use)	________mA/point
3	Electric condition	Rupture voltage	DC/AC________V
		Withstand test voltage	DC/AC___ V, ____min
		Allowable common mode voltage	DC/AC____V, ____min
4	Input rate (Response)	Repeat sampling speed	__________kHz
		Filter	YES_____No______

	Classification	Item	Description
4	(Cont.)	Filter time constant	______msec
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer time	______ /word
6	Safety function	Isolation Type	YES______No____
		Signal floating required	YES______No_____
		Validity check Type	YES______No_____
		Protection circuit Type	YES______No_____
7	Structure (Configuration)	Connections Type Terminals size	
		Module unit Card Unit Enclosure	
		Max. no of input point	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Input Circuit	(shown by drawing)	

Description of Digital Output (Form 6)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Output characteristics	Output signal level	
		High level (Make)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Low level (Break)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Contact rating (contineous)	min____V, max____V
			min____A, max____A
			_______VA
		Contact rating (Instantaneous)	_____V_____msec
			_____A_____msec
			_____VA_____msec
		Holding time of switch	
		Time adjustment (Type)	Software, Hardware
		Kinds and accuracy of time	_____ms± _____ms
			_____ms± _____ms
			_____ms± _____ms
			_____ms± _____ms

	Classification	Item	Description
2	(Cont.)	Method of adjustment	FIX; Semi-fix; Variable
		Power source	
		Internal	DC_____V ± _____V _______mA/point
		External	DC_____V ± _____V _______mA/point
		Mechanical life time	
3	Electric condition	Rupture voltage	DC/AC________V
		Withstand test voltage	DC/AC______V, ____min
		Allowable common mode voltage	DC/AC______V, ____min
4	Output rate (Response)	Repeat sampling speed	________kHz
		Contact-bounce	
		0 1	________msec
		1 0	________msec
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer time	_______ s/word
6	Safety function	Isolation Type	YES______No_____
		Load floating required	YES______No_____
		Internal protection circuit	
		External protection circuit	

	Classification	Item	Description
6	(Çont.)	Condition at power failure recovery	Hold Reset Instability
		Validity check	
7	Structure (Configuration)	Connections Type Terminals size	
		Module unit Card Unit Enclosure	
		Max. no of output point	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Out Circuit	(shown by drawing)	

Description of Pulse Train Input (Form 7)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Input characteristics	Input signal level	
		High level (Make)	
		Logical	0 : 1
		Load (Contact)	_____to_____
		" (Electronic)	_____to_____V
		" (")	_____to_____mA
		Low level (Break)	
		Logical	0 : 1
		Load (Contact)	_____to_____
		" (Electronic)	_____to_____V
		" (")	_____to_____mA
		External contact rating	
		Power source	
		Internal	DC_____V ±_____V
			_____mA/point
		External	DC_____V ±_____V
		(Internal use)	_____mA/point
3	Electric condition	Rupture voltage	DC/AC__________V
		Withstand test voltage	DC/AC_____V, _____min
		Allowable common mode voltage	DC/AC_____V, _____min
4	Input rate (Response)	Repeat input rate	__________kHz
		Filter	YES_____ No_____
		Allowable contact-bounce	__________ms
		Make ratio	__________%

	Classification	Item	Description
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer time	
		Counter Type Size Presetting	
6	Safety function	Isolation Type	YES______No_____
		Signal floating required	YES______No_____
		Validity check Type	YES______No_____
		Protection circuit Type	YES______No_____
		Data protection	
7	Structure (configuration)	Connection Type Terminals size	
		Module unit Card Unit Enclosure	
		Max. no of input point	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Input Circuit	(shown by drawing)	

Description of Pulse Train Output (Form 8)

	Classification	Item	Description
1	Purposes	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Output characteristics	Output signal level	
		High level (Make)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Low level (Break)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Contactrating (Contineous)	min____M, max____V
			min____A, max____A
			______VA
		Contactrating (Instantaneous)	______V______msec
			______A______msec
			_____VA______msec
		Output load selection	
		Selection	YES______No_____
		Parallel selection	YES______No_____
		No. of load	
		Method	
		Selection of polarity	YES______No_____
		Type	

	Classification	Item	Description
2	(Cont.)	Frequency of pulse	
		Method of adjustment	
		Kinds	
		Cycle Time	__________ms
		Max. no of pulse	
		Power source	
		Internal	DC_____V ± _____V _____mA/point
		External	DC_____V ± _____V _____mA/point
		Mechanical lifetime	
3	Electric condition	Rupture voltage	DC/AC________V
		Withstand test voltage	DC/AC_____V, ____min
		Allowable common mode voltage	DC-AC_____V, ____min
4	Output rate (Response)	Contact-bounce	YES______, No______
		Contineous	________ms
		Up	________ms
		Down	________ms
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer time	______ s/pulse
		Counter	YES______No_____
		Size	________Bits
6	Safety function	Isolation	YES______No_____
		Type	

	Classification	Item	Description
6	(Cont.)	Load floating required	YES______No_____
		Internal protection circuit	
		External protection circuit	
		Validity check	
7	Structure (Configuration)	Connections Type Terminal size	
		Module unit Card Unit Enclosure	
		Max. no of output points	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Output Circuit	(shown by drawing)	

Description of Pulse Width Output (Form 9)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Type of Terminal Unit	
2	Output characteristics	Output signal level	
		High level (Make)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Low level (Break)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Contact rating (Contineous)	min____V, max____V
			min____A, max____A
			_______VA
		Contact rating (Instantaneous)	_____V______msec
			_____A______msec
			_____VA_____msec
		Output load selection	
		Selection	YES_____No_____
		Parallel selection	YES_____No_____
		No. of load	
		Method	

	Classification	Item	Description
2	(Cont.)	Selection of polarity	YES______No______
		Type	
		Base pulse width	
		Type	
		Kinds	
		Time	________msec
		Maximum pulse width (per one action)	________msec
		Power source	
		Internal	DC_____V ± _____V _______A/point
		External	DC_____V ± _____V _______A/point
		Mechanical lifetime	
3	Electric Condition	Rupture voltage	DC/AC______V
		Withstand test voltage	DC/AC_____V, _____min
		Allowable common mode voltage	DC/AC_____V, _____min
4	Output rate (Response)	Contact bounce	YES______No_____
		Contineous	________msec
		Up	________msec
		Down	________msec
5	Operational mode	Transfer control mode	
		Transfer block length	
		Transfer time	________msec
		Counter	YES______No_____
		Size	________Bits

	Classification	Item	Description
6	Safety function	Isolation	YES______No_____
		Type	
		Load floating required	YES______No_____
		Internal protection circuit	
		External protection circuit	
		Validity check	
7	Structure (Configuration)	Connections	
		Type	
		Terminals size	
		Module unit	
		Card	
		Unit	
		Enclosure	
		Max. no of output point	
8	Optional		
9	Block Diagram	(shown by drawing)	
10	Output Circuit	(shown by drawing)	

Description of Interrupt Input (Form 10)

	Classification	Item	Description
1	Purpose	Name of Computer to be applicable	
		Basic type of Application	
		Name of Operating System (OS)	
2	Input characteristics	Max. no. of Interrupt Level	________Level
		Max. no. of Interrupt Point "	Hardware_____points Software______points
		Input signal level	
		High level (Make)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Low level (Break)	
		Logical	0 : 1
		Load	_____to_____V
		"	_____to_____mA
		"	_____to_____
		Interrupt trigger	
		Rising edge	
		Falling edge	
		Contact rating	________V
			________mA
			________VA
		Power source	
		Internal	DC_____V ±_____V
			________A/point

	Classification	Item	Description
2	(Cont.)	External	DC_____V ±_____V ______ A/point
3	Electric condition	Rupture voltage	DC/AC______V
		Withstand test voltage	DC/AC_____V, ____min
		Allowable common mode voltage	DC/AC_____V, ____min
4	Input rate (Response)	Response time	
		Total Response Time	________msec
		Hardware portion	________msec
		Software portion	________msec
		Repeat input rate	________Hz
		Minimum pulse width	
		High level	________msec
		Low level	________msec
		Contact boune	________msec
5	Operational mode	Assignment of interrupt level	
		Interrupt points each level	
		Hardware	______P'ts/level
		Software	______P'ts/level
		Reference of Interruptive factor	Hardware, Software
		Priority interrupt handling	
		Masking of Interrupt	
		Testing of Priority	Hardware, Software
		Save registers	Hardware, Software
		Method	

	Classification	Item	Description
5	(Cont.)	Generation of branch address	Hardware, Software
		Restortion of registers	
		Time of Interrupt handling after signal received	________msec
6	Safety function	Isolation Type	YES ______No______
		Signal floating required	
		Validity checking	
		Protection circuit	
7	Structure (Configuration)	Connections Type Terminals size	
		Module unit Card Unit Enclosure	
		Unit of additional level	
		Max. no of points	
8	Optional		
9	Block Diagram for procedure	(shown by drawing)	
10	Interrupt Circuit	(shown by drawing)	

Comments for Description of Process Input/Output Interface

	Item	German Proposal	Japanese Proposal
1	Composition of standard proposal	Vocabulary Analog Input Analog Output Digital Input Digital Output Interrupt ISO/TC 97/SC 13 (Doc.No. 97/13N 49 55)	General (on Common) Analog Input Analog Output Analog Control Output Digital Input Pulse Train Input Digital Output Pulse Train Output Pulse Width Output Interrupt (Doc.No. JEIDA 49-A-82)
2	Usage	More information for Hardware engineer	More information for User or System planner
3	Vocabulary	DEFINED	NOT YET
4	Definitions	(Require more detail definitions of performance and dynamic characteristics.)	
5	Relationale	(Scope of interface) (Related industries's requirements to on-line computing system) (Grade of definition parameters)	
6	Common definitions	Not defined	Add the follows; -- System configuration of general -- Power supply condition -- Grounding condition -- Cable specification -- Structure -- Installation

	Item	German Proposal	Japanese Proposal
7	Digital Input		
7.1	Proposal	97/13 N 51 (Digital Input)	Consist of two drafts ° Digital Input ° Pulse Train Input
7.2	Classification	-- Electronic Input -- Contact Input	Same form
7.3	Performance Mode (5.1)	-- Static Input -- Dynamic Input -- Pulse Input	-- Static Input (Defined as Digital Input) -- Pulse Input (Defined as Pulse Train Input) -- No Dynamic Input (This function is defined Special function as Status change detector.)
7.4	Explosion Proof and PTB-License	Defined	Not defined
7.5	Pulse Shape	Defined in 5.7 as Pulse shape.	Defined by Filter, Time constant and Repetitive speed.
7.6	Technology (4.4)	Defined	Not defined (which component to be specified)
8	Interrupt Input		
8.1	Proposal	97/13 N52 (Interrupt Input)	Interrupt Input
8.2	Classification	-- Electronic Input -- Contact Input	Same form

	Item	German Proposal	Japanese Proposal
8.3	Time Relations	According the definition of characteristic time intervals in 4.1	Define the characteristics of response till First trap condition in CPU (include hardware and software handling time).
8.4	Interrupt Level	Require more detail definition	Same
8.5	Explosion Proof and PTB-License	Defined	Not defined
8.6	Pulse Shape	Defined in 5.7 as Pulse Shape	Defined by the characteristics of response
8.7	Architecture (4.6)	Defined	Not defined (which component to be specified)
9	Digital Output		Consist of three drafts
9.1	Proposal	97/13 N 53 (Digital Output)	° Digital Output ° Pulse Train Output ° Pulse Width Output
9.2	Classification	-- Electronic Output -- Contact Output	Same form
9.3	Functional characteristics	-- Static Output -- Pulse Output (Duration) -- Pulse Output (Frequency)	-- Static Output (Defined as Digital Output) -- Pulse duration output (Defined as Pulse Width Output) -- Pulse frequency output (Defined as Pulse Train Output)

	Item	German Proposal	Japanese Proposal
9.4	Logical signal characteristics		Define Momentary output characteristics Define Transient rating of Output
9.5	Explosion Proof and PTB-Licence	Defined	Not defined
9.6	Architecture (4.6)	Defined	Not defined (Which component to be specified)
9.7	Pulse duration		-- Defined selectivity of output load -- Defined polarity of pulse -- Defined standard pulse
9.8	Pulse string		-- Defined selectivity of output load -- Defined polarity of pulse -- Defined standard pulse
9.9	Capacitive and Inductive Load	Defined	Not defined
10	Analog Input		
10.1	Proposal	97/13 N 54	Analog Input
10.2	Accuracy		Add the follows. ° Linearity Error ° Repeatability
10.3	Common mode rejection	Defined CMR (dB)vs. frequency	Defined CMR (ISA-RP 55.1)

	Item	German Proposal	Japanese Proposal
10.4	Crosstalk		Defined crosstalf of -- Common mode -- AC -- DC
10.5	Functional characteristics (5.1)	Defined the function of ADC	Defined by the follows -- Filter -- Multiplexer -- Amplifier -- ADC
10.6	Signal Type	Defined by the follows -- Voltage input -- Current input	Defined by the follows -- Voltage input -- Current input -- Resistance input -- Thermocouple input -- Special input
10.7	Explosion Proof and PTB-License	Defined	Not defined
10.8	Architecture ·	Defined	Not defined (Which component to be specified)
10.9	Settling Time	Defined	Defined by Filter time constant
11	Analog Output		
11.1	Proposal	97/13 N 55	Consist of two drafts ° Analog Output ° Analog Control Output
11.2	Accuracy		Add the follows ° Droop Rate

	Item	German Proposal	Japanese Proposal
11.3	Signal Type	Defined by the follows -- Voltage Output -- Current Output	Defined by the follows -- Range -- Rating
11.4	Explosion Proof and PTB-License	Defined	Not defined
11.5	Architecture	Defined	Not defined (Which component to be specified)
11.6	DDC use	Not defined	Defined in Analog Control Output (Backup capability)

SERIAL LINE SHARING SYSTEM

FOR

INDUSTRIAL REAL-TIME APPLICATIONS (SIR)

Prepared by:

TC5, Interfaces and Data Transmission
Purdue, Europe.

August 1976

Contents of the System Draft

1. Introduction

With the increasing size and complexity of industrial processes the provision of centralised control systems, based on a conventional star structure, is becoming uneconomic. Dezentralised Systems are therefore tending towards a bit-serial line-sharing approach with a common bus interconnecting many stations.

An essential feature of such systems is communication between geographically distributed subsystems within the plant. That aspect of the total system which provides the communication facilities is referred to here as "the communication subsystem". The communication subsystem has a number of access points at which it interacts with the other subsystems requiring or generating information. These access points are referred to as "stations". A "Master Station" is the access point at which the control of the communication subsystem is exercised. It is not necessarily the source of control for the complete system. A "Slave Station"is an access point which responds to the Master Station. The subsystem connected at a station (whether it be a source of control at a Master Station or responding equipment at a Slave Station) is referred to as a "device subsystem" (or briefly as a "device") in order to distinguish it from the communication subsystem.

Any communication subsystem is a compromise between speed of response, reliability and cost. The particular trade-off selected is related to the specific environment and is a matter for the system designer. It may well result in different communications subsystems being employed within a single plant.

A communication subsystem should be capable of incorporation within a hierarchical structure. Figure 1 shows, in diagramatic form, the various levels which might require a communication subsystem. At level A the subsystem provides communication between the computer exercising overall control (or its stand-by) and the computers dedicated to specific processes. Particularly vital information may be extracted at this level from specific device subsystems constituted as one or more clusters. It should be noted that a cluster is simply an economic method of connecting a number of separate units to the communication subsystem at a single station. At level B the particular process control computer communicates with its device subsystems (which may be further process control computers). In this hierarchy the "Process X Control" computer is connected at a Slave station to the level A communication subsystem, and is connected as a source of control at the Master Station of the level B communication subsystem. This structure is open to higher and lower levels of automation hierarchy but is also applicable to a single level in a simple scheme. The provision of a communication adaptor between levels in parallel with the process control computer permits a higher level computer to assume the functions of a lower level computer in the event of failure. Such a facility requires a degree of compatibility between the separate communication subsystems at the two levels.

In addition to the problem of defining an appropriate communication subsystem, the designer of a specific application system may find that the increasing specialisation of manufacturers of automation equipment makes it difficult to procure components compatible with a common bus.

The goal of this proposal is therefore to define the general characteristics of line-sharing communication subsystems appropriate to the process control environment, and to propose standard methods of interconnection between the communication subsystem and the attached device subsystems. Two defined ports providing this interconnection have been identified. One is at the interface between the communication subsystem and the device subsystem, and is independent of the technology and protocols of either subsystem. The other port is within the communication subsystem and is at the interface between the line dependent part and the remainder of the communication subsystem. The choice of which of these defined ports should be implemented physically in a specific application is still under discussion in the committee.

The proposal first outlines the application areas and overall requirements for process control applications (section 2). It then makes recommendations on the general structure of a communication subsystem (section 3). This is followed by a description of the two types of defined ports (section 4). Finally, methods of implementation are discussed and a preferred communication subsystem,conforming to the requirements and using the defined ports, is described (section 5). The preferred communication subsystem also serves to illustrate the previous recommendations.

2. Application Areas and System Requirements

This proposal relates to communication subsystems for process control within an industrial plant. Computer-to-Computer communication or the connection of standard peripherals is not the primary aim of the proposal, but the proposal gives a form of these facilities appropriate to process control as a consequence of its general approach.

A process control system is an on-line real-time system in which reaction times are important and overall reliability is essential. The characteristic which differentiates process control communication subsystems from other on-line real-time systems is that the output causes material or energy to move. This necessitates secure and usually dedicated channels, and implies an in-plant cable system installed expressly (but not necessarily exclusively) for process control signals.

Typical application areas are:

Power generation stations

Oil industry

Testing and quality control of engines

Chemical plants

Steel plants

Manufacturing Systems (DNC)

2.1 Requirements

The general requirements for a communications subsystem are:

1. It should provide reliable and economic bit-serial communication between device subsystems such that a command message may be sent to, and a reply message received from, another subsystem without involving store and forward operations.

2. Within the communication subsystem stations are provided for the attachment of other subsystems. A source of commands uses an access point known as a Master Station. Access-points for subsystems which respond to these commands are known as Slave Stations. At any one time only one Master Station is permitted in the communication subsystem.

3. With the increasing complexity of computer based process control systems it is highly desirable that means be provided for transferring Mastership from one station to another. Three major examples of this are:

(a) Stand-by control facilities:

The normal Master position is duplicated at a designated position so that if there is a failure the duplicate may take over the role of Master in a pre-planned manner.

(b) Servicing and test facilities:

Each station is able to accept equipment which has the capability (to a greater or lesser extent) of performing Master functions for a limited period as a test facility. This may require prior permission.

(c) Direct data-interchange between stations:

Each station may incorporate equipment which has the dynamic capability of assuming Mastership in order to communicate directly with any other station. The transfer of Mastership would be an inbuilt routine feature of the communication subsystem.

4. The communication subsystem should be capable of being incorporated in a hierarchic or single level structure in centralised or distributed control systems.

5. The communication subsystem must be capable of working within a noisy environment with a low relative rate of undetected errors and an appropriate throughput of valid messages. The subsystem design should permit enhancement of the error detection performance in critical applications.

6. The reliability and availability of a particular implementation will depend on the designer's choice of equipment quality and noise protection. The intent of this proposal is that the system should incorporate error detection facilities which may be matched to the environment in which the system is used. Desired are 1oooo hours at least for a station (see Figure 3) excluding the device(s).

7. The installation or removal of equipment at a station is permitted to disturb the current messages as a transitory effect provided that the system is able to detect such disturbances and recover full operation within a time appropriate to the application.

8. The communication subsystem must be designed such that communication is maintained between a Master and a Slave in spite of:

(a) Power loss at another Slave station

(b) Failure of the equipment at another Slave station. (It should be noted that this failure can result in noise being generated which must be prevented from masking traffic).

9. The communication subsystem design should allow redundant paths between stations to be incorporated when required to give enhanced reliability. This should permit the system to continue operation (with perhaps reduced performance) after the loss of one or more paths between stations.

10. The system overhead for small installations should be minimised.

11. The interface between the communication subsystem and device subsystems must be independent of the particular technology (cable, radio-links, light-pipes etc.) used within the specific communication subsystem, and should be independent of the communication protocol (error detection, demand-handling etc.).

12. The communication subsystem should be transparent to distance considerations, when viewed from its stations. In present practice a typical distance between Master and Slave stations or between Slave stations is 300-1000metres. With a loop configuration the average total bus line length is expected to be 1.5 Km with an upper figure of 5Km. Provision must be made for the communication subsystem to make use of a common carrier when necessary, for example when crossing a public highway.

13. The design should permit galvanic isolation to be provided between the communication subsystem and the devices mounted at stations.

14. The information should be conveyed in serial binary digital form within an appropriate structure. Measurement and control parameters often require an accuracy better than 0.1% and hence the binary digital representation of a parameter value may require in excess of 10 bits. A general recommendation is a minimum of 12-bits for value plus one-bit for sign (that is, a total of 13-bits or 14-bits if a parity bit is included). In a byte-oriented system a minimum of 2 bytes should be used.

2.2 Typical Data, some examples of present industrial plants

(See Table on page 59)

3. General Recommendations for a Communication Subsystem

The following sections give general recommendations for a bit-serial line-sharing communication subsystem. It is assumed that, at any given time, there is one predefined and fixed Master Station in the communication subsystem. In an extension, planned as future work, the effect of transferring Mastership dynamically will be investigated. It may be, but cannot be guaranteed, that a system with transferrable Mastership will form a superset with the current recommendations.

Communication is by a "Command Message" from the Master Station to any one of many Slave Stations; and by a "Reply Message" from the addressed Slave Station to the Master Station. The subsystem may also permit a global Command Message to be transmitted from the Master Station to more than one Slave Station.

In a communication subsystem with "Active" Slave Stations activity may be initiated by an Active Slave Station generating a "Demand Request" when it requires service.

3.1 Master Station

The Master Station controls the communication subsystem. It is often, but not necessarily, linked to the device subsystem exercising process monitoring and supervisory control of the overall system. The functions of the Master station are:

(a) to generate Command Messages and transmit them to Slave stations served by the communication subsystem.

(b) to receive Reply Messages from the Slave stations in response to Command Messages, or to detect the absence of a solicited Reply Message, and take appropriate action.

(c) to accept Demand Requests from Slave stations (in a communication subsystem with Active Slaves) and take appropriate action.

3.2 Slave Station

A Slave station conforms to the communication subsystem procedures and protocols generated by the Master. The functions of a Slave station are:

(a) to identify, accept and implement valid Command Messages received from the Master station.

(b) to generate an appropriate Reply Message to every valid Command Message individually addressed to it. (The response to Global Command Messages is the subject of on-going work, but it is currently assumed that global commands do not solicit a reply).

Field	Input			Output		Timing	
	Analogue	Binary	Binary Alarm	Analogue	Binary	Scan Time	Mir. Reaction-time *
Power generation stations	2ooo	6ooo	6oo	8oo	2oo	2...12o sec	1oo msec
Oil industry	16o	18oo	5o	-	12oo	1...12o sec	1oo msec
Testing and quality control of engines	1oo	5oo	1oo	25	2oo	o,o1-1o sec	1o msec
Chemical plants	4oo	5oo	3oo	5o	6oo	o.1...6o sec	1oo msec
Steel plants	1oo	5oo	5o	5o	1oo	o,5...6o sec	1o msec
Manufacturing Systems (DNC)	-	5oo	5oo	-	2oo	-	1o msec

* Reaction-time is defined as the time between two events. The first event is a process alarm, the second event is the respond back to the process.

(c) to generate a Demand Request when service is required (if the Slave is an Active Slave). This demand may occur asynchronously (i.e. at any time) at the Slave station but may be delayed in transmission to the Master station (e.g. it may be delayed in accessing the line by existing traffic, or may have to wait for a specific operation).

3.3 The Structure of Messages in the Communication Subsystem

A complete transaction in the communication subsystem is the successful transmission of a Command Message to a Slave station and the receipt of a valid Reply Message at the Master station. Both Command and Reply Messages contain all information relevant to the transaction and do not, for example, require preliminary messages to establish the route.

There are three major aspects of these messages which may be distinguished (see the example in Figure 2).

3.3.1 Communication signalling and framing

The actual line signals used by the communication subsystem are a function of the communication technology (cable, radio-link, light-pipe etc.) Similarly the signalling technique and protocol determine the synchronisation signals defining the beginning and end of a message, and any internal framing used (for example, Start and Stop signals framing individual bytes).

These signals relate solely to the communication subsystem and are not passed to or from the attached device subsystems.

3.3.2 Message Control Information

This includes all aspects of the message required for identification, routing and error-detection within the communication subsystem. The information is conveyed in 8-bit fields in order to simplify generation and acceptance. It should be noted that certain fields are generated or acted upon by the device subsystems at Master and/or Slave stations.

The internal structure of Command and Reply Messages is the same and makes use of the following message control fields.

(i) Address field (ADDR) 8-bits

The address field contains the binary representation of the address of the Slave station. It is the destination address in a Command Message and the source address in a Reply Message. The 'all ones' and 'all zeroes' addresses are reserved for test functions. Some of the remaining 254 addresses may be reserved for special purposes, e.g. global commands addressed to all stations.

(ii) Message identification field (IDENT) 8-bits

(a) Fixed/variable length subfield 1-bit

The bit identifies a fixed or variable length for the device dependent information in the message. Fixed length means a predetermined length known to the system. It is an implementation option and length zero is not excluded. In general the predetermined length will be constant for a given communication subsystem. In more sophisticated applications the length may be specific to the station identified in the address field and/or have different values for Command and Reply Messages.

If variable length is specified the message includes length and routing fields (see iii and iv below).

(b) Command/reply subfield 1-bit

Since Command and Reply Messages have identical structure they are distinguished by this field which has value '1' for Command Messages and value '0' for Reply Messages.

(c) Function subfield 6-bits

This field is loaded by the Master in a Command Message and the same content is returned by the Slave in the Reply Message. It therefore serves to correlate Command-Reply transactions. It may, for example, be a message serial number used for sequence checking and error recovery purposes. It also indicates whether the message is directed to the device subsystem or is directed solely to the communication subsystem, e.g. as part of an initialisation procedure.

(iii) Length field (LENGTH) 8-bits

For a fixed length message this field is not present (see ii.a) above). For a variable length message it defines the multiple of eight bits in the device dependent information (excluding any error detection fields specific to the communication subsystem). The value of this length parameter will probably be required by both the communication subsystem (in error detection) and the device subsystem (to indicate the storage required).

(iv) Routing field (ROUTE) 8-bits

For a fixed length message this field is not present (see ii.a above). For a variable length message it gives additional information relating to the device subsystem (e.g. subaddresses). The information is not normally used by the communication subsystem.

(v) Error detection fields

The message control information should include error detection facilities, appropriate to the environment, which protect the whole message including the device dependent information. It is recommended that the address and message identification fields are protected separately from the rest of the message since they may be processed before the whole message is available. Similarly the length and routing fields are an optional pair of parameters and it may be desirable for them to have separate protection.

3.3.3 Device Dependent Information

This contains information which is specific to the particular device subsystem located at the Slave station involved in the transaction. It is conveyed as an integer multiple of 8 bits with an overall maximum length determined by the communication subsystem. Otherwise its structure and content is independent of the communication subsystem. The information content may include routing or error detection features specific to the attached device subsystem. Error detection required for message handling by the communication subsystem is provided by the message control information (see 3.3.2.v).

3.4 The Facilities Required at a Station

The facilities required at a Master or a Slave station may be divided into three parts corresponding to the three aspects of a message (see 3.3 above). These may be related to three conceptual units within the equipment at a station. These are shown in Figure 3 as

The Bus Coupler Unit

The Communication Interface Unit

and The Device Interface Unit

3.4.1 The Bus Coupler Unit

The Bus Coupler Unit is specific to the technology employed in the communication subsystem. Its operation is essentially passive in that all messages are treated identically irrespective of their content.

The functions of the Bus Coupler are:

(a) to convert between the signal standards of the communication subsystem common bus and the standard for binary signals required within the equipment at the station,

(b) to detect the beginning and end of messages received from the common bus and to provide the corresponding synchronisation signals in transmitted messages,

(c) to handle individual byte framing, if this is a feature of the communication subsystem,

(d) to provide galvanic isolation between the equipment at the station and the common bus,

(e) to provide multiple or alternative connections to the common bus, if this is a feature of the communication subsystem.

Other functions of the Bus Coupler Unit depend on whether it is employed at a Master or Slave station. At a Slave station these additional functions are:

(f) to maintain the integrity of the common bus by providing either that there are no active components in the common communication path (e.g. by transformer coupling) or that continuity is guaranteed in the event of local failure (e.g. by automatic bypassing).

(g) to provide the local clock, if the communication subsystem requires individual clocks at each station.

At a Master station the additional functions of the Bus Coupler Unit are:

(h) to provide appropriate connection to the line or lines of the common bus, with termination if required.

(i) to provide an individual clock or the common clock depending on the requirement of the communication subsystem.

If transferrable Mastership is incorporated it may prove necessary to switch between these additional functions, and to provide means for a station to request and be granted Mastership.

The intention is that the Bus Coupler Unit should be a relatively simple unit which isolates the other units at a station from the specific line technology and does not effect passing commands and replies from other stations. Signal reshaping is permitted. The Bus Coupler Unit is interconnected with the Communication Interface Unit by the Bus Coupler Port (see section 4.1).

3.4.2 The Communication Interface Unit

The Communication Interface Unit is specific to the message protocol of the communication subsystem but is independent of the line signalling technique employed. It is also independent of the characteristics of the attached device subsystem.

The functions of the Communication Interface Unit are related to its use in receiving or transmitting messages. It should be noted that a Master transmits Commands and receives Replies whereas a Slave receives Commands and transmits Replies. The functions of the Communication Interface Unit at a Slave Station are:

(a) to check received messages (Commands addressed to the station) and pass them to the Device Interface Unit.

(b) to format messages from information provided by the Device Interface Unit and transmit them (Reply Messages).

(c) to generate demands, if an Active Slave.

The corresponding functions at a Master Station are:

(a) to check received messages (Replies) and pass them to the Device Interface Unit.

(b) to format messages from information provided by the Device Interface Unit and transmit them (Command Messages).

(c) to accept demands, if Active Slaves are allowed in the communication subsystem.

3.4.3 The Device Interface Unit

The Device Interface Unit is specific to the attached device subsystem but is independent of the communication subsystem. The equipment that may be connected is virtually unrestricted except that the device subsystem at a Master station must be capable of specifying appropriate commands and interpreting the replies. Similarly the device subsystem at a Slave station must respond to valid commands and send appropriate replies.

The Device Interface Unit can be designed for individual units, clusters of units, complete subsystems or intelligent subsystems (e.g. computers and microprocessors). Thus the Device Interface Unit may include internal addressing and error detection facilities related to the attached equipment. This is totally distinct from the facilities provided by the communication subsystem.

Specific Device Interface Units may be designed to connect:

- Process devices (e.g. digital transducers) with bit-serial or bit-parallel signal coding (typically 12-bits plus sign).

- Peripherals using byte-or word-serial methods as specified in

  IEC Bus (IEC/TC 66/WG 3)
  British Standard Interface (BS 4421)
  FNI Interface for Peripheral Devices
  (FNI/AA13)
  Medical Interface proposal V1000P
  (GMDS-Germany)
  Standard Communication Interface
  (CIS-France)
  CAMAC Data Way (EUR 41oo)

- Minicomputers with a channel interface, (as currently under consideration in ISO/TC97/SC13)

- A specific microprocessor

- Specific peripheral equipment e.g. a Visual Display Unit.

3.5 The Structure of the Communication Subsystem

Following from the conceptual separation of the hardware at a station (illustrated in Figure 3) the general structure of the complete system is as shown in Figure 4. It will be seen that the boundary of the communication subsystem is at the Independent Port between the Communication Interface Unit and the Device Interface Unit at each station. At the Master station the device subsystem includes the source of commands and at each Slave station the device subsystem contains the equipment which responds to these commands.

Within the communication subsystem the Bus Coupler Ports provide the boundary to the "Line Technology Dependent Part" of the subsystem and effectively isolate the rest of the system from the signalling techniques and protocols of the line.

The fundamental purpose of the communication subsystem is to enable the source of commands to send a specific command to remote equipment and receive a reply. The specific commands and replies are regarded as device dependent information, and it is assumed that appropriate device protocols and procedures are incorporated to give useful communication. Figure 5 shows that, at the Master station, this device dependent information is augmented with message control information (in the Communication Interface Unit) and bracketed with synchronisation signals (in the Bus Coupler Unit) to form the Command Message on the common bus (see Figure 6).

At the Slave station the synchronisation is stripped by the Bus Coupler Unit, the message is identified and checked in the Communication Interface Unit, and the device dependent information is passed to the Device Interface Unit.

For a Reply Message the same sequence of events is followed but the device dependent information originates in the device subsystem at a Slave station and is delivered to the device subsystem at the Master station.

It is recognised that standardisation of the device protocols and procedures would be useful, but that is not the purpose of this proposal. The object is solely to define the mechanism by which the device dependent information (regardless of content) is transferred between the device subsystems at a Master and a Slave station.

4. The Defined Ports

The conceptual division of the equipment at a station into three units (see Figure 3) permits two ports to be identified and defined. These are the Bus Coupler Port which provides the connection between the Bus Coupler Unit and the Communication Interface Unit; and the Independent Port which provides the connection between the Communication Interface Unit and the Device Interface Unit. There may well be a third port (or ports) between the Device Interface Unit and the specific equipment. This is however device dependent and not the subject of this proposal.

4.1 The Bus Coupler Port

All information transfer through the Bus Coupler Port is either from the Bus Coupler Unit to the Communication Interface Unit (Receive Function) or from the Communication Interface Unit to the Bus Coupler Unit (Transmit Function). A Master station passes the information for a Command Message with a Transmit Function and receives the information from a Reply Message with a Receive Function. In some implementations of a closed loop system the Master may also monitor the returned command with a separate Receive Function.

A Slave station receives a Command Message by a Receive Function. However, since the Bus Coupler does not examine the information content of messages, all Command Messages (and possibly Reply Messages) appearing on the common bus at the Slave station will be passed through the Bus Coupler Port. A Reply Message from the device subsystem at the Slave station is passed to the Bus Coupler Unit with a Transmit Function.

The information is passed as a bit stream on a data line with an accompanying strobe signal on a second line and a control signal on a third line for each direction of transfer (see Figure 7). The message content is conveyed by an integer multiple of 8-bits, and (under certain circumstances) there can be simultaneous reception and transmission. An example of the information transfer mechanism is given in Figure 8.

4.1.1 Receive Function

The Receive Function is performed by three signals generated in the Bus Coupler Unit (see Figures 7 and 8).

(i) Receive Message Present (RMP)

This signal is generated by the Bus Coupler Unit and is maintained for the duration of a message. Its initiation is interpreted as the beginning of a message and instructs the Communication Interface Unit to accept information for identification and checking. Any previous incomplete message in the Communication Interface Unit is abandoned. The removal of the signal indicates the end of the message and triggers the response in the Communication Interface Unit, e.g. in the case of a Slave, if the message is valid the action requested is implemented and a Reply Message generated.

The Bus Coupler Unit generates the 'Receive Message Pres signal by detecting the beginning and end of the message fro message framing of the communication subsystem; that is, it responds to the "Beginning of Message" and "End of Message" synchronisation signals. It is noted that a distinction bet these two synchronisation signals within the line protocol reduces the problems of phasing should a spurious synchronisa signal be encountered.

(ii) Strobe (S)

The 'Strobe' is generated by the Bus Coupler Unit is derived from the line clock of the common bus. When the 'Receive Message Present' signal is asserted the 'Strobe' si indicates that the 'Receive Data' signal is staticised as a binary zero or one and should be accepted by the Communicati Interface Unit.

When the 'Transmit Message Present' signal (see 4.1.2) asserted the Strobe signal indicates that a 'Transmit Data' is required.

In certain implementations of demand handling 'Strobe' signals may be generated by the Bus Coupler when neither 'Rec Message Present' nor 'Transmit Message Present' is asserted. The 'Strobe' is then interpreted by the Communication Interfa Unit as a request for the status of Demand Request.

(iii) Receive Data (RD)

The 'Receive Data' signal is generated by the Bus Coupler Unit and is staticised at either binary one or zero f acceptance within the period of the 'Strobe' signal. Its v is that of the corresponding bit in the information content the message; synchronisation signals and byte-framing Start Stop signals (if used) are not transmitted. The Receive Da information is conveyed as a bit stream containing an intege multiple of eight bits.

4.1.2 Transmit Function

The Transmit Function is performed by three signa generated in the Communication Interface Unit and makes use the Strobe signal (4.1.1.ii) from the Bus Coupler (see Figur and 8).

(i) Transmit Message Present (TMP)

This signal is generated by the Communication Interface Unit and is maintained for the duration of a messa The Bus Coupler interpretes the initiation of 'Transmit Mess

Present' as a request to transmit a message and as an indication that the information content for the message is available. The Bus Coupler may therefore proceed with the generation of a "Beginning of Message" synchronisation signal without storing within itself the complete message. Such an approach is not debarred if required by the communication subsystem.

While 'Transmit Message Present' is asserted the Bus Coupler requests each individual data bit by transmitting a 'Strobe' signal. This is an essential feature since only the Bus Coupler is aware of the timing required by the line protocol of the communication subsystem.

(ii) Transmit Strobe (TS)

The 'Transmit Strobe' is generated by the Communication Interface Unit in response to the 'Strobe' signal from the Bus Coupler Unit. The 'Transmit Strobe' indicates that the signal 'Transmit Data' is staticised as a binary zero or one and should be accepted by the Bus Coupler Unit.

The main function of the 'Transmit Strobe' signal is to reduce timing errors between the 'Strobe' (from the Bus Coupler Unit) and 'Transmit Data' (from the Communication Interface Unit) by providing a timing signal from the same source as the data. The physical distance permitted between the Bus Coupler Unit and the Communication Interface Unit is a function of the time delay allowed between the transmission of the 'Strobe' and the acceptance of the 'Transmit Strobe' at the Bus Coupler Unit. This is itself a function of the line protocol on the common bus and whether or not buffering is provided in the Bus Coupler Unit.

When 'Transmit Message Present' is asserted the 'Transmit Strobe' indicates the presence of individual information bits forming part of a Command or Reply Message.

When 'Transmit Message Present' is not asserted the 'Transmit Strobe' may indicate the presence of individual bits (as 'Transmit Data' signals) related to demand handling.

(iii) Transmit Data (TD)

The 'Transmit Data' signal is generated by the Communication Interface Unit and is staticised at either binary one or zero for acceptance within the period of the 'Transmit Strobe' signal.

When 'Transmit Message Present' is asserted its value is that of the corresponding bit in the information content of a message (a Command Message at a Master station or a Reply Message at a Slave station). Synchronisation signals and byte framing Start and Stop signals (if used) are not generated in the

Communication Interface Unit and hence are not passed through the Bus Coupler Port. The Transmit Data information is conveyed as a bit-stream containing an integer multiple of eight bits.

When 'Transmit Message Present' is not asserted the value of the 'Transmit Data' signal relates to demand handling.

4.1.3 Operation of the Bus Coupler Port

Figure 8 shows an example of the operation of the Bus Coupler Port at a Slave station. In this example it is assumed for simplicity that Command and Reply Messages are physically separated (for example on separate lines). This is not an essential feature of the proposal.

In order to identify the beginning and end of a received message the Bus Coupler Unit must recognise the message synchronisation signals of the line protocol. It is assumed that these synchronisation signals each occupy an equivalent time to 'n' information bits in the message, and that the Bus Coupler Unit includes a delay of n bits in the information path to the Bus Coupler Port. In the figure n has been arbitarily made equal to 4 for simplicity. No delay is introduced into the signal paths on the common bus.

The Bus Coupler Unit recognises the synchronisation signal appearing on the common bus. Depending on the line protocol this can convey different levels of information. In the example it is assumed that the signal is identified as the beginning of a Command Message; however in a system with Reply and Command Messages on the same line the message type may not be distinguished. Equally with only a single version of the synchronisation signal the distinction between the beginning and end of a message may rely on context.

The message content is passed to the Communication Interface Unit as a bit stream on the 'Receive Data' line accompanied by individual 'Strobe' signals and an overall 'Receive Message Present' signal. Identification of the "End of Message" synchronisation signal causes 'Receive Message Present' to be removed. In the example the delay in the path to the Bus Coupler Port enables the "End of Message" synchronisation to be recognised without being passed through the port. The delay also allows the 'Strobe' signal to be separated from the common bus line clock (although derived from it); for example as shown, effectively delayed by eight bits. Alternatively the strobe may control the data transfer as a sequence of high speed bursts of bits while remaining compatible with the line clock rate. If no delay is incorporated the removal of 'Receive Message Present' at an appropriate time may present difficulties.

With an implementation in which two lines are available on the Common Bus the Communication Interface Unit can generate demand handling information by sending 'Transmit Strobe' and 'Transmit Data' signals while a message is being received. Alternatively, or in addition, the Bus Coupler Unit can request demand handling information by sending a 'Strobe' within a defined time of removing 'Receive Message Present'. The defined time ensures that the Communication Interface Unit has not asserted 'Transmit Message Present'.

It is a function of the Communication Interface Unit to identify the message as a Command Message addressed to itself either at the end of the message or before. All other messages are ignored.

If a reply is to be generated the Communication Interface Unit generates 'Transmit Message Present'. The Bus Coupler sends the synchronising signal for "Beginning of Reply Message" (or its equivalent) to the common bus and then requests invididual bits of the message with the 'Strobe' signal. These 'Strobe' signals may be derived directly from the line clock, may be effectively in front of the line clock (in order to prestore the information as illustrated) or may be sent as bursts at a higher frequency. The method used influences the time delay that can be tolerated in obtaining the required information and hence the physical distance that can be allowed between the Bus Coupler Unit and the Communication Interface Unit.

Removel of 'Transmit Message Present' causes the Bus Coupler Unit to append the "End of Message" synchronisation to the message on the common bus.

Figure 9 illustrates the operation of the Bus Coupler Port at a Master Station. The operation is very similar to that at a Slave station except that the Transmit Function applies to a Command Message and the Receive Function to a Reply Message. In a two-line closed-loop implementation demand handling information generated during a Command Message must be monitored against the Command returned after traversing the loop. This requires an additional Receive Function capability at the Master station.

Demand handling information generated between a Command Message and a Reply Message can be transmitted to the Communication Interface Unit at the Master station by the Bus Coupler unit sending 'Strobe' and 'Receive Data' signals when neither 'Receive Message Present' nor 'Transmitt Message Present' is asserted.

4.2 The Independent Port

The Independent Port provides the interconnection between the Communication Interface Unit and the Device Interface Unit (see Figure 3). Information transfer is either from the Communication Interface Unit to the Device Interface Unit (Receive Function) or from the Device Interface Unit to the Communication Interface Unit (Transmit Function). Since the Receive and Transmit Functions are virtually the same for the Independent Port as for the Bus Coupler Port it is recommended that they be performed by six signals analogous to those defined for the Bus Coupler Port.

The Communication Interface Unit processes the information contained within messages and as a consequence there are additional control lines at the Independent Port. The signals used by the Independent Port are illustrated in Figure 10 and examples are given of their use at a Slave Station (Figure 11) and at a Master Station (Figure 12).

4.2.1 Receive Function

At a Master station all Reply Messages are passed to the Device Interface Unit by the Communication Interface Unit. At a Slave station those Command messages addressed to the station and directed to the device subsystem are passed to the Device Interface Unit. The Receive Function is performed by five signals generated in the Communication Interface Unit (see Figures 10, 11 and 12).

(i) Receive Message Present (RMP)

This signal is generated in the Communication Interface Unit and is maintained for the duration of the message. Any previous incomplete message in the Device Interface Unit is abandoned.

(ii) Strobe (S)

The 'Strobe' is generated in the Communication Interface Unit. When the 'Receive Message Present' signal is asserted the 'Strobe' signal indicates that the 'Receive Data' signal is staticised as binary zero or one and should be accepted by the Device Interface Unit. When the 'Transmit Message Present' signal is asserted the 'Strobe' signal indicates that a 'Transmit Data' bit is required.

(iii) Receive Data (RD)

The 'Receive Data' signal is generated by the Communication Interface Unit and is staticised at either binary one or zero for acceptance within the period of the 'Strobe' signal. Its value is that of the corresponding bit in the message. Error detection information specific to the communication subsystem is not transmitted. The Receive Data information is conveyed as a bit stream containing an integer multiple of eight bits.

(iv) Receive Length (RL)

During a Receive Function the Communication Interface Unit examines the Message Identification Field to determine whether the message is of fixed or variable length. If the message is of variable length the Communication Interface Unit asserts the 'Receive Length' signal.

(v) Receive Error Detected (RED)

During a Receive Function the Communication Interface Unit checks the message content using the error detection protocol of the communication subsystem. Should an error be detected this signal is set by the Communication Interface Unit and the total message content is ignored by the Device Interface Unit.

4.2.2 Transmit Function

The Transmit Function makes use of the Strobe signal generated by the Communication Interface Unit (see 4.2.1.ii) and four signals generated by the Device Interface Unit (see Figures 10, 11 and 12).

(i) Transmit Message Present (TMP)

This signal is generated by the Device Interface Unit and is maintained for the duration of the message. The Communication Interface Unit interprets the initiation of 'Transmit Message Present' as a request to transmit a message and as an indication that the information content for the message is available.

While 'Transmit Message Present' is asserted the Communication Interface Unit requests each individual data bit by transmitting a 'Strobe' signal (see 4.2.1.ii).

(ii) Transmit Strobe (TS)

The 'Transmit Strobe' signal is generated by the Device Interface Unit in response to the 'Strobe' signal from the Communication Interface Unit. The 'Transmit Strobe' signal indicates that the 'Transmit Data' signal is staticised at binary one or zero and should be accepted by the Communication Interface Unit.

(iii) Transmit Data (TD)

The 'Transmit Data' signal is generated by the Device Interface Unit and is staticised at either binary one or zero for acceptance within the period of the 'Transmit Strobe' signal. Its value is that of the corresponding bit in the information content of the message. The error detection required by the communication subsystem is not generated in the Device Interface Unit and hence is not passed through the Independent Port. The Transmit Data is conveyed as a bit-stream containing an integer multiple of eight bits.

(iv) Transmit Length (TL)

During a Transmit Function the Device Interface Unit asserts a 'Transmit Length' signal if the message is of variable length. The information generated by the Device Interface Unit then includes the optional field which defines the message length.

4.2.3 Demand Handling

Demand handling is a function of the communication subsystem and the Communication Interface Unit conforms to the subsystem protocol. The only information required from the Device Interface Unit at a Slave station is that a demand is present. At a Master station the Communication Interface Unit can indicate the presence of a demand within the system by generating 'Demand Present'. Further information (e.g. a binary number specifying a particular demand) may be passed as data-bits on the 'Receive Data' line while 'Demand Present' is asserted (if the communication subsystem has this facility).

The signal 'Demand Present' is therefore generated at a Slave station by the Device Interface Unit and is maintained until the demand is satisfied in accordance with the device protocol. At a Master station 'Demand Present' is generated by the Communication Interface Unit and is accompanied by such additional information as is provided by the communication subsystem.

4.2.4 Operation of the Independent Port

Figure 11 shows an example of the operation of the Independent Port at a Slave station. In this example it is assumed that the functions of the Bus Coupler Unit are perfomed separately (identification of beginning and end of messages, handling of synchronisation and framing signals). However the definition of the Independent Port does not require the Bus Coupler Unit and the Communication Interface Unit to be physically separable (for example, by implementation of the Bus Coupler Port).

The first two fields of a message are the 8-bit Address field and the 8-bit Message Identification field. The action of the Communication Interface Unit on receiving a message is therefore to check these fields (using whatever error-detection is provided by the communication subsystem) and, if error free, to act on the content. Only Command Messages addressed to the Slave and directed to the device subsystem are passed through the Independent Port. The first two fields are not required by the device subsystem and hence are not transmitted; but the information in the fixed/ variable length subfield sets the 'Receive Length' signal at the time the 'Receive Message Present' signal is generated.

If a variable length message has been identified, the next two fields to be received are the 8-bit Length field and the 8-bit Route field. These are passed through the port by the 'Receive Data' and 'Strobe' signals, the length information being also stored in the Communication Interface Unit for use in error checking.

The device dependent information in the message is passed to the Device Interface Unit after the error detection required by the communication subsystem has been removed. Theoretically it may seem desirable for the entire Command Message to be guaranteed error-free before any part of it is passed to the Device Interface Unit. However this would require the Communication Interface Unit to incorporate sufficient storage to hold the longest possible message. In practice the desired result is achieved by transmitting the information content through the port as it is received and providing a 'Receive Error Detected' signal from the Communication Interface Unit to the Device Interface Unit. Provided that the information received is as expected (e.g. it is of correct length and conforms to the format and error codes for the equipment) AND 'Receive Error Detected' has NOT been received the Device Interface Unit may act on the information. The message is terminated by removing the 'Receive Data Present' signal.

With a Reply Message from a Slave the Device Interface Unit sends 'Transmit Message Present' and 'Transmit Length'. The

Communication Interface Unit may then send the first two fields (Address and Message Identification) of the message to line. The Communication Interface Unit requests further information as individual bits by sending the 'Strobe' signal. The Device Interface Unit responds with 'Transmit Data' and 'Transmit Strobe' signals.

If a variable length has been specified the 8-bit Length and 8-bit Route fields are passed to the Communication Interface Unit for onward transmission. The length information is also stored by the Communication Interface Unit so that for either fixed or variable length messages the appropriate error detection fields may be added.

The Device Interface Unit passes the device dependent information and then removes "Transmit Message Present'.

The Device Interface Unit may indicate that it requires service by asserting the 'Demand Present' signal at any time. It remains set until the request is satisfied.

Figure 12 illustrates the use of the Independent Port at a Master station. The operation is very similar to that at a Slave station except that the Transmit Function applies to a Command Message and the Receive Function to a Reply Message.

The Device Interface Unit initiates a Command Message by setting the 'Transmit Message Present' and 'Transmit Length' signals. The Communication Interface Unit requests individual bits of the message with 'Strobe' signals and the Device Interface Unit responds with 'Transmit Data' and 'Transmit Strobe'.

The Address and Message Identification fields, the Length and Routing fields (if variable length), and the device dependent information fields (if present) all originate from the Device Interface Unit. The Communication Interface Unit adds the appropriate error detection fields before passing the message for transmission to line.

All messages received at the Master station are passed to the Device Interface Unit with the communication-subsystem error-detection fields removed. The 'Receive Error Detected' signal gives the required assurance of validity.

The message is initiated by setting 'Receive Message Present' and the information content is passed by the 'Receive Data' and 'Strobe' signals.

In principle, for a system in which each transaction is completed before the next is begun, the Address and Message Identification fields of a Reply Message could be processed in the Communication Interface Unit by comparison with the corresponding fields of the Command Message. However such an approach might limit the error recovery capability of the system, and hence both these fields are passed to the Device Interface Unit.

The receipt of the Message Identification field at the Device Interface Unit makes the 'Receive Length' signal redundent at a Master station. It is however retained for symmetry.

After the transmission of the Length and Routing fields (if variable length) and the device dependent information (if present) the Communication Interface Unit terminates the message by the removal of 'Receive Message Present'. Provided that the information is as expected (e.g. it is of correct length and conforms to the device protocol) AND 'Receive Error Detected' has NOT been received the Device Interface Unit may act on the information.

Demands received at the Master Station may be passed to the Device Interface Unit at an appropriate time by generating the 'Demand Present' signal. Additional information, for example, identifying the specific demand or demands can be conveyed by the 'Receive Data' and 'Strobe' signals while 'Demand Present' is asserted.

4.3 Physical Implementation of the Defined Ports

The conceptual division of the hardware at a station into three units (see Figure 3) allows two ports to be defined. The Bus Coupler Port is independent of the specific technology used by the communication subsystem (cable, radio, light-pipe etc.) but is dependent on the communication protocol (demand-handling, error-detection etc.). The Independent Port is independent of the protocol and procedures of both the communication subsystem and the device subsystem at the station.

The use of the defined interfaces simplifies the interconnection of equipment from different manufacturers and is particularly relevant to systems which may require modification or extension. The decision on whether either or both of the defined ports should be physically implemented on a specific system is a matter for the system designer. It is noted that the Independent Port gives flexibility of connection of device subsystems at Slave stations (irrespective of the communication subsystem selected) but may be of lower value at the Master station (which is probably less liable to change). The Bus Coupler Port gives independence of line technology and hence may prove of equal value at both Master and Slave stations. The implementation of both ports results in the Communication Interface Unit having totally defined interconnections. A unit conforming to an agreed protocol can then be constructed for general application and may lead to quantity production and reduced costs.

For the implementation of either port the EIA standard RS422 "Electrical Characteristics of Balanced Voltage Digital Interface Circuits" is recommended (see Figure 13). EIA RS422 is also published as SP-1162-A and as CCITT-X27.

Compared with unbalanced signals EIA RS422 has the following advantages in process control application.

(i) The interconnecting twisted pair cable is less sensitive to noise.

(ii) Fail safe operation is provided.

(iii) Longer distances are possible.

(iv) Cross talk is reduced.

(v) Signal inversion may be achieved by reversing the cable pair.

The maximum recommended cable length is a function of the transmission rate. The transmitter has to generate a low impedence (100 ohms) balanced differential voltage in the range 2 to 6 volts (see Figure 13).

If a physically separable connection is provided it is recommended that this make use of the 25-way Cannon type DBC-25P, single density fixed member with pins, or equivalent. Equivalent connectors include AMP-Minrac 17-series 17-10250-1, and Cinch-D*SM, DBSM-25P.

The pin assignment is to be agreed.

5. Applications of the Defined Ports

The ports defined in this proposal have been carefully selected to impose the minimum constraints on the communication subsystem or on any attached device subsystems. Thus the device dependent information is virtually unrestricted by the port definitions. For example the Device Interface Unit may format the information received from the common bus as bit-stream, byte-structure or word-structure as required by the interfaced equipment. The information content may also include specific formatting bits and error detection. Thus seven bit ASCII with odd-parity added is a simple example of device dependent coding.

Similarly the communication subsystem has a minimum protocol imposed by the ports. It may define its own synchronisation and framing techniques and use whatever timing mechanism is appropriate. It will be noted that the 'Strobe' signal is, of necessity, generated on the communication side of each port (since information must be generated and accepted at the line rate) but that buffering may be provided between the line clock and the ports.

5.1 Typical Communication Subsystems

A number of possible communication subsystems have been investigated. These can be subdivided into closed loop and open line systems; and into one and two line systems (see Figure 14).

In a closed loop system the signal path originates at a Master station, threads through all Slave stations and returns to the Master. All messages are transmitted in the same direction round the loop and a Command Message is received at the Master where it may be compared with the original transmission if required.

In an open line system the Command Message is sent out and in due course a Reply Message is received. The Master may be at the end of the line or inserted in the line with propogation in both directions.

In a two line system a common line clock may be provided by the Master on one line and Reply Messages inserted on the other line by the addressed Slave. The Command Message may also be on the Reply Line, or combined with the clock on its line. With these techniques active components may be excluded from the lines. All access is then by transformer coupling and no electronic delays are incurred.

In a one line system there is a choice of technology. Either active components must be inserted in the line in order to demodulate and remodulate the central system clock at each station, or separate clocks must be provided at each station. The first technique gives potential failure if the power supply at a station is lost and hence automatic bypassing must be incorporated, while the second involves reducing traffic to allow for settling times and resynchronisation of clocks between successive messages.

5.2 A Preferred Communication Subsystem

A preferred implementation of the communication subsystem uses a two line closed loop. Command Messages and a continuous clock are provided as a combined signal on one line (using, for example, Manchester Biphase Modulation). Reply Messages and demand handling information are conveyed on the other line. The common line clock provides the timing information. This method uses transformer coupling as the means of access to the line (see Figure 15).

Alternative line transmission methods (e.g. single line with separate clocks at each station) can be used within the preferred implementation since the Bus Coupler Port effective isolates the line technology from the rest of the subsystem. Beginning and end of message synchronisation signals depend on the transmission method employed.

Figure 16 shows the information crossing the defined ports. Figures 17 and 18 show the use of the ports in greater detail at a Slave and a Master Station respectively.

5.2.1 Error Detection

Command Messages and Reply Messages have the same format and are structured as a sequence of 8-bit units. The total message is protected by the use of cyclic redundancy checks (CRC) at appropriate points. The CRC used is the 8-bit BCH polynomial having the value

$$x^8 + x^2 + x + 1$$

The general structure of a message is as previously defined (see section 4.3) and uses the defined fields for message control information (see 4.3.2). The first field of the message is the 8-bit Address, giving the destination for a Command and the source for a Reply. This is followed by the 8-bit Message Identification which distinguishes between Command and Reply, and between fixed and variable length device dependent information. These two field are protected by a CRC field and hence may be processed and acted upon before the complete message has been received.

For a variable length message the next two fields are the 8-bit Length field and the 8-bit Route field. The length field specifies the multiple of eight bits conveying device dependent information, in the range 1 to 256. (Note, the binary value in the length field is in the range 0 to 255). The length information is used by the Communication Interface Unit in the generation and checking of the error detection fields. It is also used by the Device Interface Unit. The routing information is not used in the communication subsystem. These two fields are protected by a CRC field. For a fixed length message the Length, Route and CRC field are omitted.

The device dependent part of the message is unconstrained by the communication subsystem protocol except that it must contain an integral multiple of eight bits less than or equal to 256. The number of 8-bit units B may be represented by

$$B = h + kN \leqslant 256$$

where: N is selected for the implementation and may have value 2, 4, 8 or 16.

k is a positive integer

h is a positive integer less than or equal to N.

Within a message a CRC field is inserted after the first N 8-bit units and again after each sequence of N 8-bit units. This is a simple matter if the length parameter in the message (expressed as a binary number) is counted down during the transmission or reception of the device dependent information. Modulo 2, 4, 8 or 16 (depending on the value selected) can then be readily identified and the CRC inserted or checked at appropriate points.

By this technique the device dependent information may be conveyed by any number of 8-bit units from 1 to 256 with a known minimum error protection. This may be selected as a trade-off against transmission efficiency over the common bus. With N = 2 the transmission efficiency for a maximum length message is 65% (i.e. 256/390) and with N = 16 it is 92% (i.e. 256/278).

A Slave must not generate error messages on detecting a transmission error. For example, in a system which is based on a loop carrying Command Messages through all Slaves back to the Master, a correct message may be corrupted part-way round the loop. This can cause a normal Reply Message from the correctly addressed station to be overwritten by an Error Message from a Slave addressed as a result of the error.

From the system point of view an invalid Command Message which is detected and rejected is the same as an undelivered Command Message, and will be detected at the Master by time-out on the Reply or, in a Loop System, by examination of the returned Command Message.

5.2.2 Demand Handling

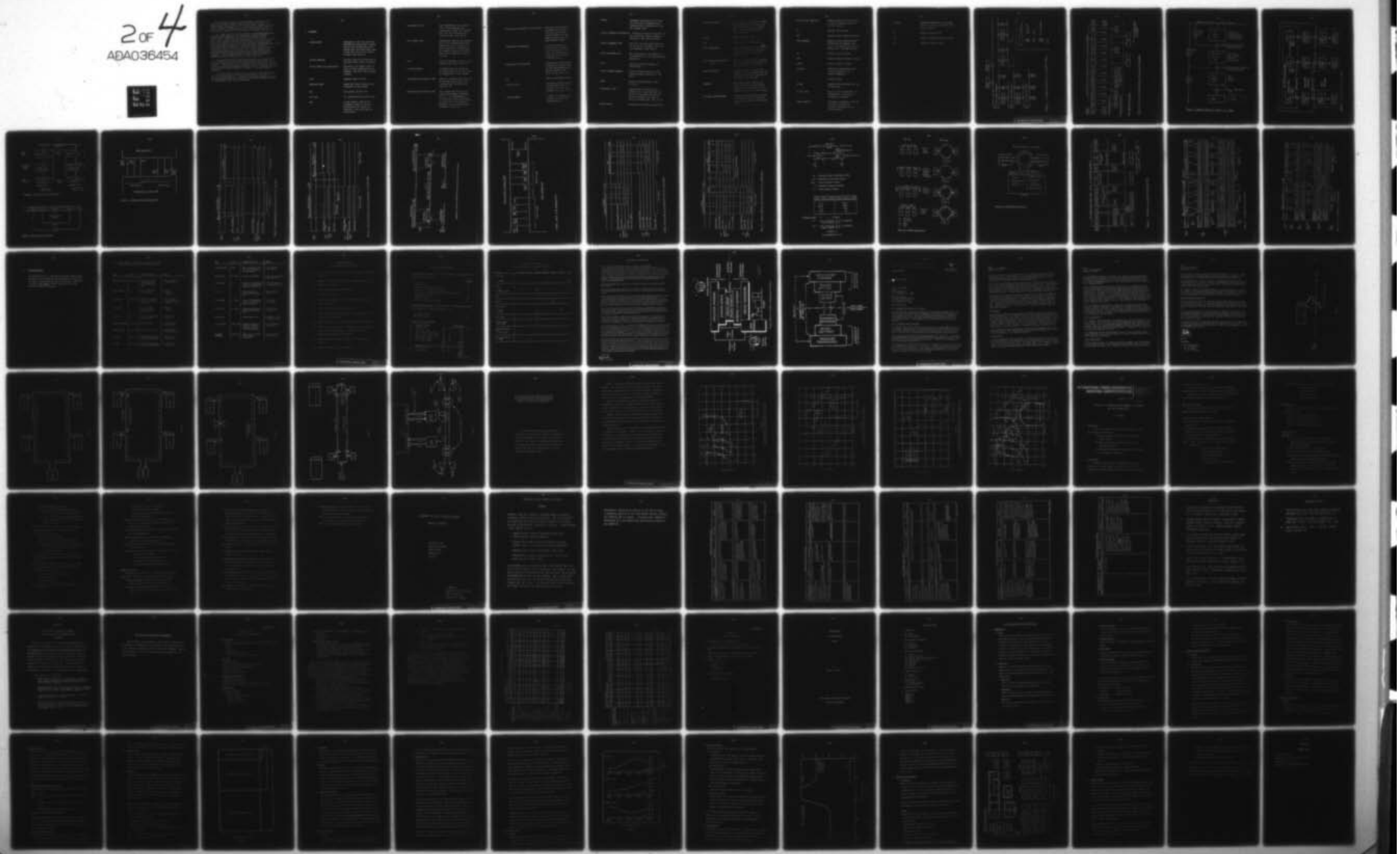
For an implementation of the preferred communication subsystem with Active Slaves, two levels of demand handling are provided (see Figures 17 and 18).

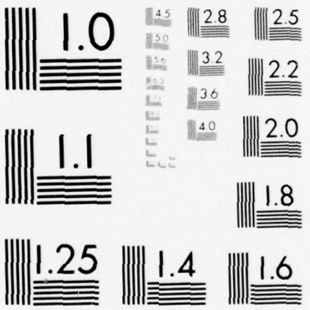
At a Slave station the Device Interface Unit may request service at any time by the signal 'Demand Present' at the Independent Port. As a first level of demand handling the Bus Coupler may request the status of this line from the Communication Interface Unit by a Strobe signal between the Transmit and Receive Functions of a transaction. Any (or all) of the Active Slaves may thereby put a signal on the common bus.

AD-A036 454 PURDUE UNIV LAFAYETTE IND PURDUE LAB FOR APPLIED IND--ETC F/G 9/2
SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIO--ETC(U)
JAN 77 N00014-76-C-0732

UNCLASSIFIED NL

2 OF 4
ADA036454




1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

At the Master station the Bus Coupler responds to this global signal with 'Receive Data' and 'Strobe' to the Communication Interface Unit which then passes 'Demand Present' to the Device Interface Unit. This level of demand handling does not allow a specific demand to be identified, but is applicable to one or two line and open or closed loop systems.

The second level of demand handling is normally applied to two line closed loop systems (where skew effects between lines can be neglected), and is used in addition to the first level. The minimum length possible for a Command Message is 24 bits (Address, Message Identification and CRC fields). Upto 24 different 1-bit demands may therefore be sent to the Reply Line during receipt of a Command Message (see Figure 9a). In an application the Communication Interface Units of upto 24 different Active Slaves may each be assigned a unique number in the range 1 to 24. If the Slave has a demand present the Communication Interface Unit marks the corresponding bit by counting 'Strobe' signals while 'Receive Message Present' is asserted. The "global demand" is also marked and gives an element of error detection to the demand handling.

At the Master station the demand signals on the Reply Line must be passed to the Communication Interface Unit while the returned Command Message is also passed by an additional Receive Function. The Communication Interface Unit can then pass on the 24-bit message to the Device Interface Unit as a Receive Function controlled by 'Demand Present' in place of 'Receive Message Present'.

It should be noted that, with the defined Independent Port, the Device subsystem is totally uninfluenced by the type of demand handling provided. However, the signals passing through the Bus Coupler Port will depend on the demand handling protocol and compatible units must be provided.

6. Glossary

Access point

Equipment on the bus, by which information interchange occours. Within the communication subsystem only one acces point may at any one time act as Master Station /See Station/.

Active coupling

Coupling mode of the devices to the line using active elements.

Active slave /or substation/

Substation /or Slave/ which is able to reply immédiately on a Command, and which may generate Demands.

ADDR

Address Field /8 bits/

Balanced cable

Symmetrial cable, which is not connected to the ground.

BCH

One special cyclic code.

Bit code

Bit representation on the line.

Bus

Signal line/s/ used by the interface system to which a multiplicity of devices is connected, and which carries information.

Bus Coupler Port

Line Independent Bus Coupler Port, which provides the connection between Bus Coupler Unit and the Communication Interface Unit /see Fig.3/

Bus Coupler Unit

Operational passive unit connected to the communication subsystem bus. It is specific to the Transmission technology, and doesn't effect the command and reply signal path through the bus.

Byte

Group of adjacent binary digits, usually consisting of 8 bits.

Command Message

A message which is generated by the master station and which is transmitted to the slaves.

Communication Dependent Part

Part of a sation which consists of the Bus Coupler Unit and Communication Interface Unit /see Fig. 3/

Communication Interface Unit

Unit placed between the bus coupler and device interface units. It is independent of the line signalling technique, but specific to the message protocol of the communication subsystem /see Fig.3/

Communication signalling and framing	Technique and procedure to control the information flow on the Bus. /Synchronisation of messages, start stop signals between bytes/
Communication Subsystem	That system part which provides the communication facilities on the line. The communication subsystem has a number of access points.
Communication Technology	Information transmission procedure and media used by the communication subsystem /cable, radio link, light pipe, etc./
CRC	Cyclic Redundancy Check.
Defined Ports	There are two identified and defined ports: Bus Coupler Port and Independent Port /see Fig.3/
Demand Request	Request for service, generated by the active slave stations.

Device
Equipment connected to the line via the Device Interface Unit, Communication Interface Unit and Bus Coupler Unit.

Device Dependent Information
Information which is specific to the particular device subsystem located at the slave station.

Device Dependent Part
Part of a station which consists of the Device Interface Unit and the Device/s/ /see Fig.3/.

Device Interface Unit
Unit connected to the device, it is independent of the communication. subsystem.

Field
Specific logical grouping of information.

Global Command Message
Common command message for all slave stations connected to the line.

IDENT
Message identification field /8 bits/.

Independent Port
Communication and Device Independent Port, which provides the connection between the Communication Interface Unit and the Device Interface Unit /see Fig.3/

Intelligence
Information processing capability.

Interface System

Set of cables, connectors, signal lines, descriptions, timing and control conventions, etc.required to effect communication among stations.

LENGTH

Length field /8 bits/.

Line

Coax cable or twisted pairs.

Line technology

Signalrepresentation and mode of information transfer on the line /type of modulation, logical levels, etc./

Line Technology Dependent Part

Part of a station which consists of the communication Bus and the Bus Coupler Unit /see Fig.3/

Master Station

Control station at which the control of the communication subsystem is executed. Generally it is linked to a computer.

Message

No interruptable sequence of bits. Frames of several bytes containing information for the Master Sation of for the Slaves.

Message Serial Number

Contentof the function subfield /see IDENT/ used for sequence checking and error recovery purposes.

Passive line coupling	Coupling mode with no galvanic connection between the line and the station.
RD	Receive Data signal.
RED	Receive Error Detected signal.
Reply Message	Message which contains information for the Master, and which is generated by a Slave-station on a Command Message.
RL	Receive Length signal.
RMP	Receive Message Present signal.
ROUTE	Routing field /8 bits/.
Routing	Particular information flow control related to the device subsystem.
S	Strobe signal
Signal	Physical representation of the information.
Signal line	Set of signal conductors for transferring informations between the stations.
Slave Station	Equipment connected to the line, which responds to commands generated by the current Master.

Station	Equipment connected to the line which is able to communicate with other Stations.
TD	Transmit Data signal.
TMP	Transmit Message Present signal.
TS	Transmit Strobe signal.

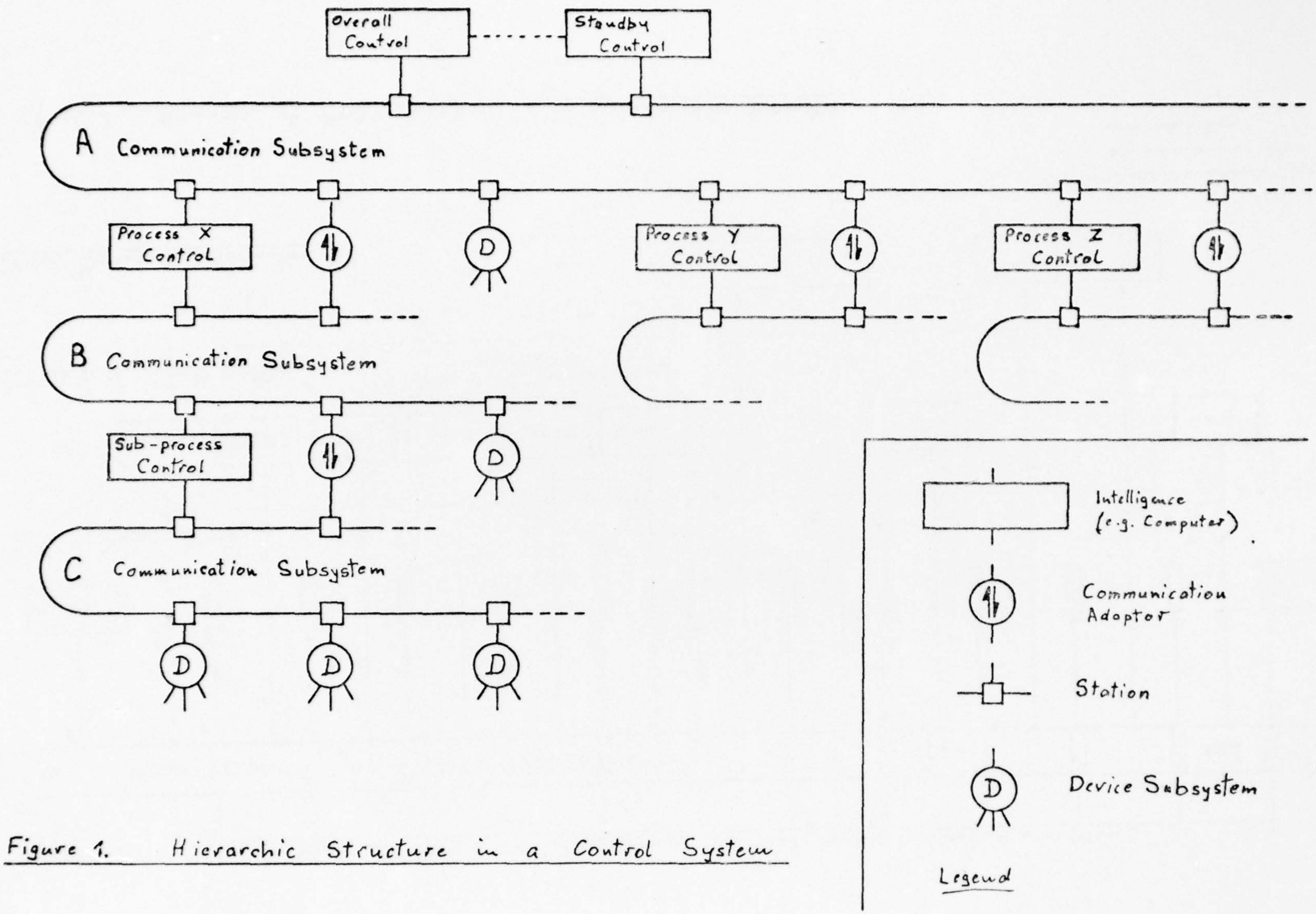
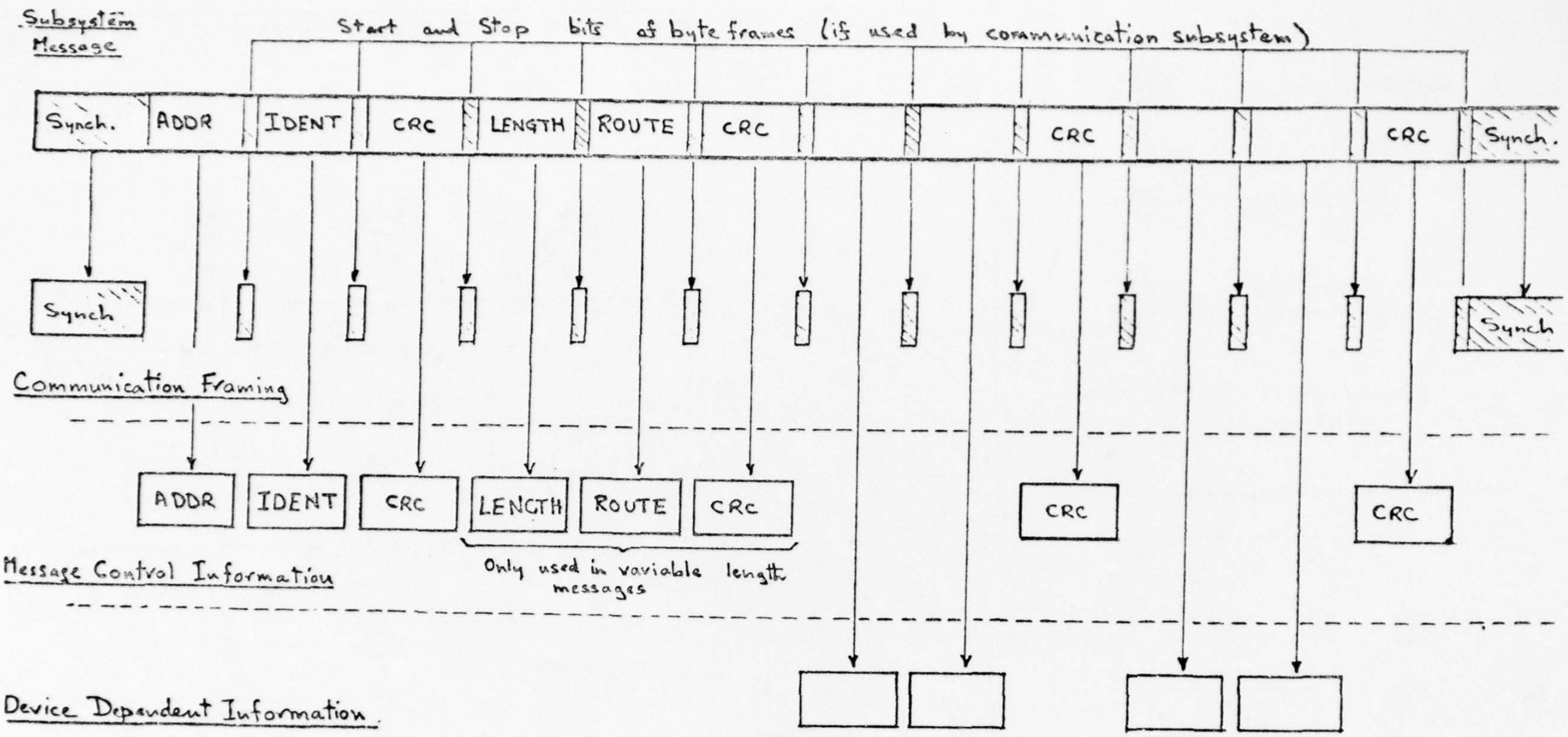


Figure 1. Hierarchic Structure in a Control System


Subsystem Message
Start and Stop bits of byte frames (if used by communication subsystem)
Synch.
ADDR
IDENT
CRC
LENGTH
ROUTE
CRC
CRC
CRC
Synch.
Synch
Synch
Communication Framing
ADDR
IDENT
CRC
LENGTH
ROUTE
CRC
CRC
CRC
Only used in variable length messages
Message Control Information
Device Dependent Information

CRC = Cyclic Redundancy Check as an example of error detection.

Figure 2 Example of Message Content in the Communication Subsystem.

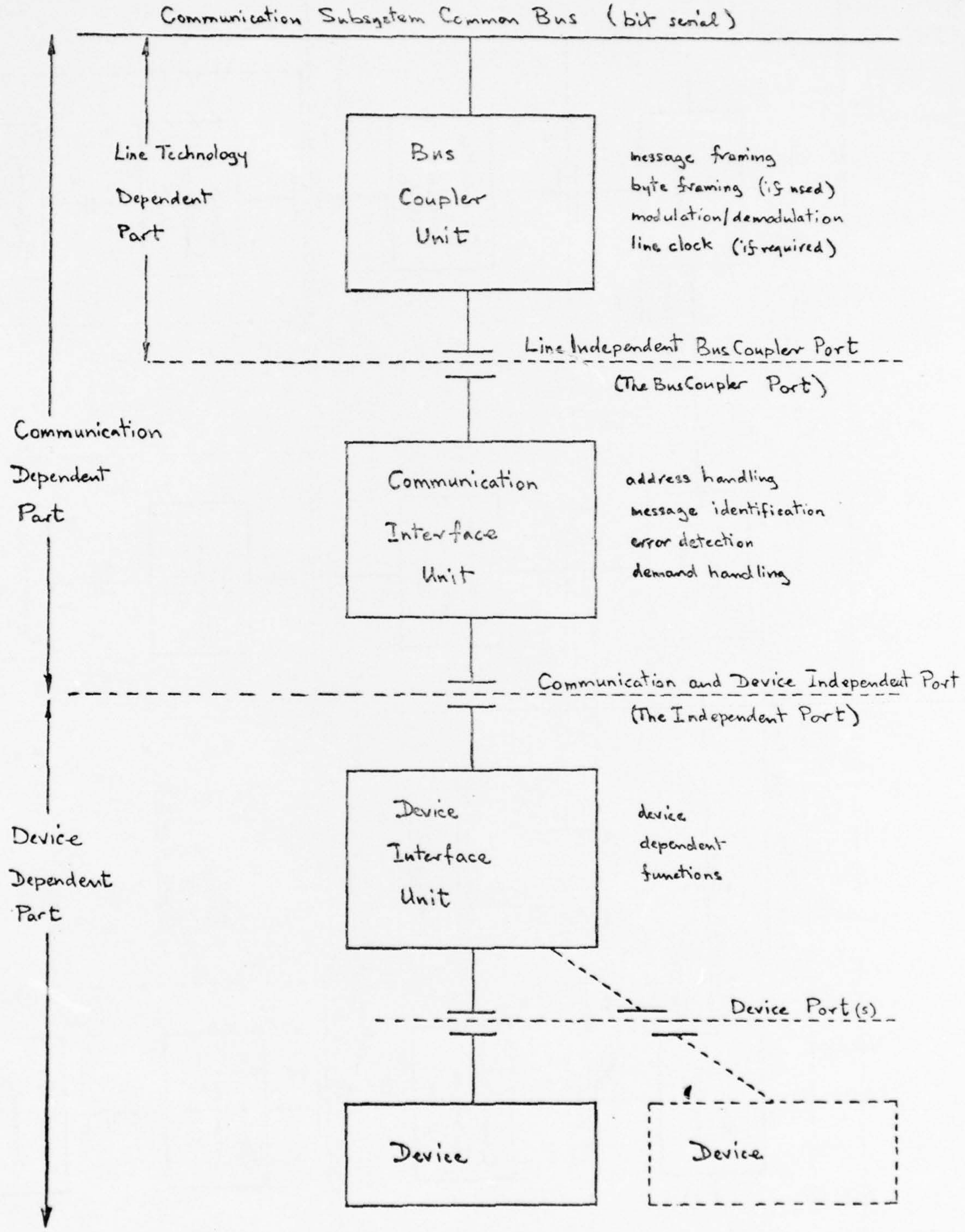


Figure 3, Conceptual Division of Hardware at a Station

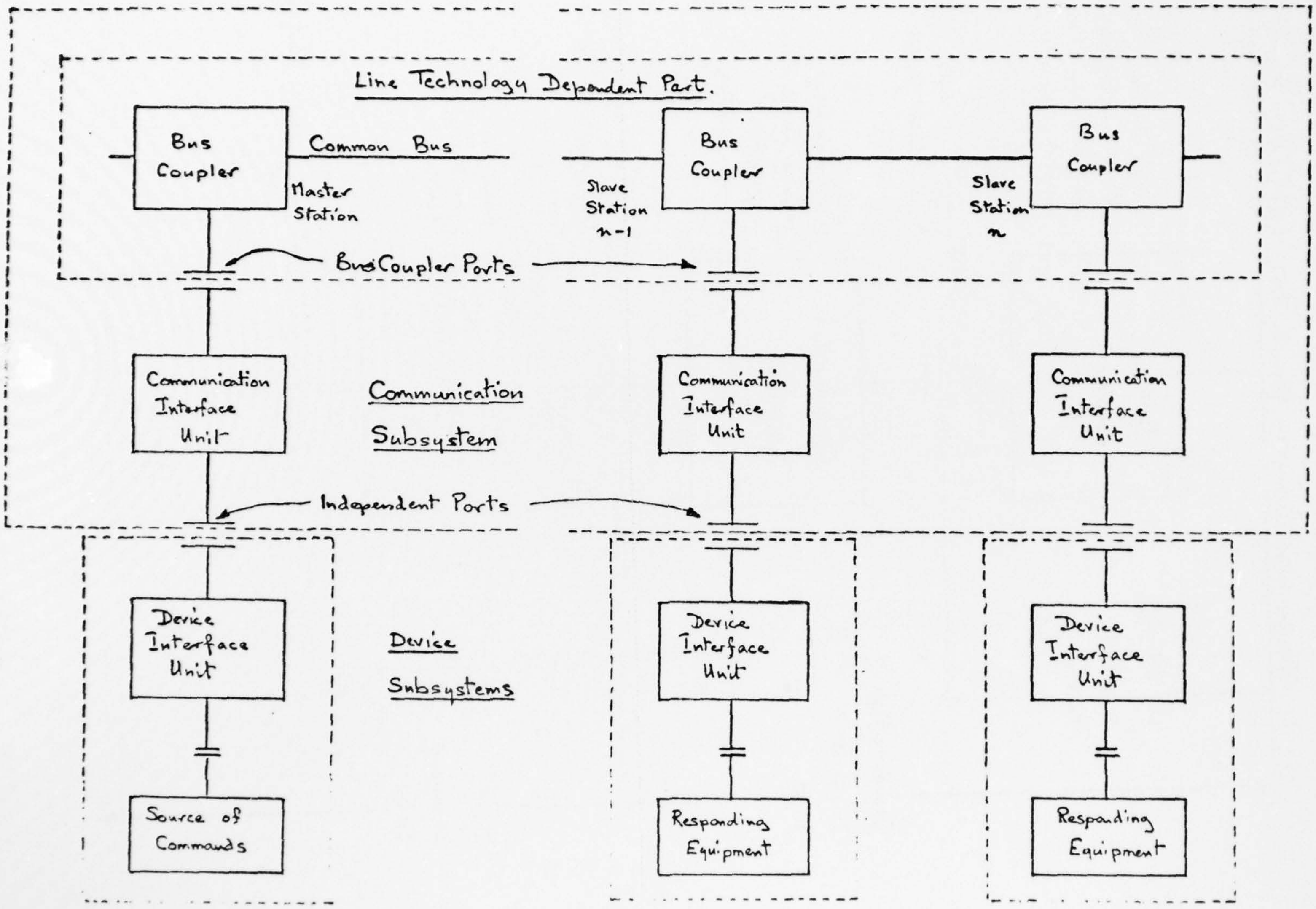


Figure 4. Structure of a Process Control System.

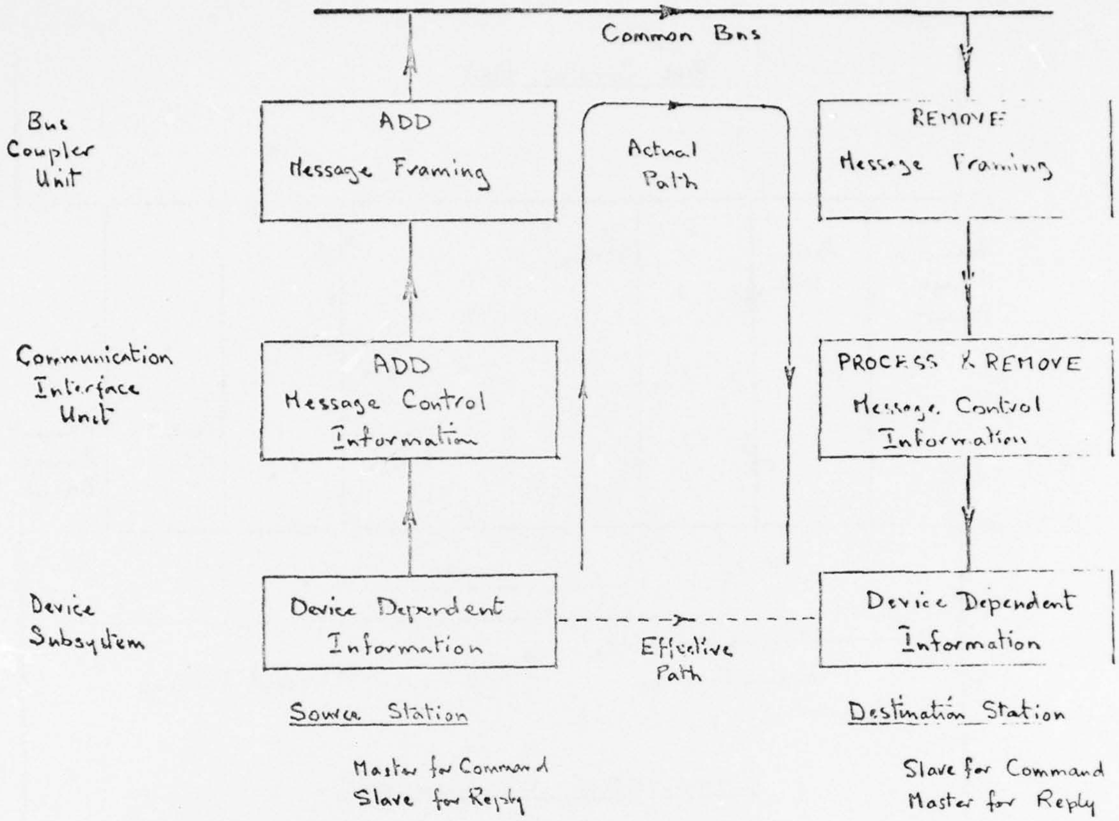


Figure 5. Information Flow through the System

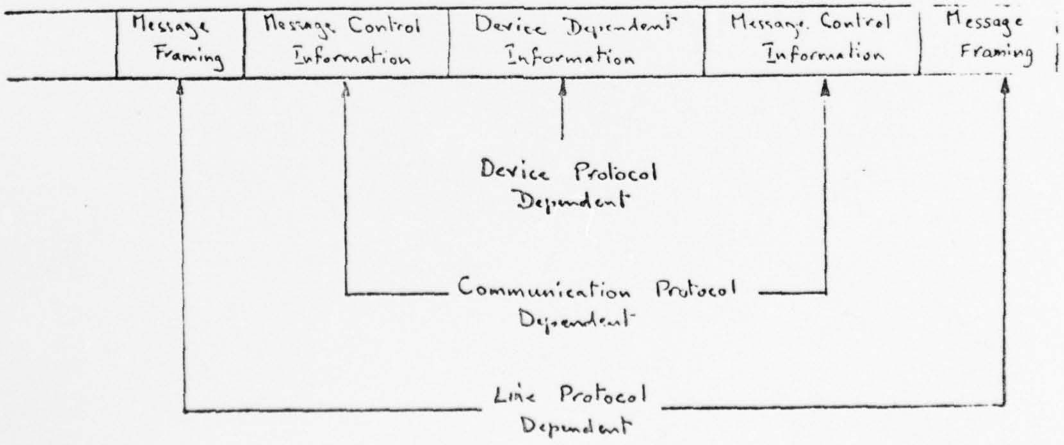

Message Framing
Message Control Information
Device Dependent Information
Message Control Information
Message Framing
Device Protocol Dependent
Communication Protocol Dependent
Line Protocol Dependent

Figure 6. Schematic Message Structure

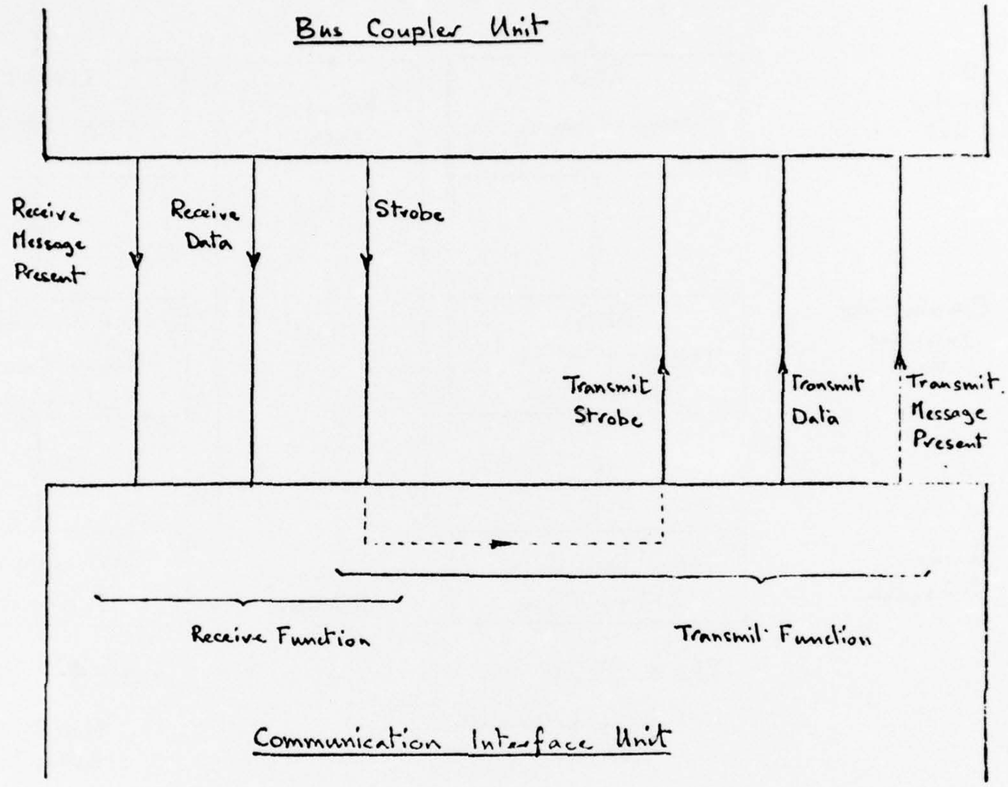


Figure 7. Signals at the Bus Coupler Port

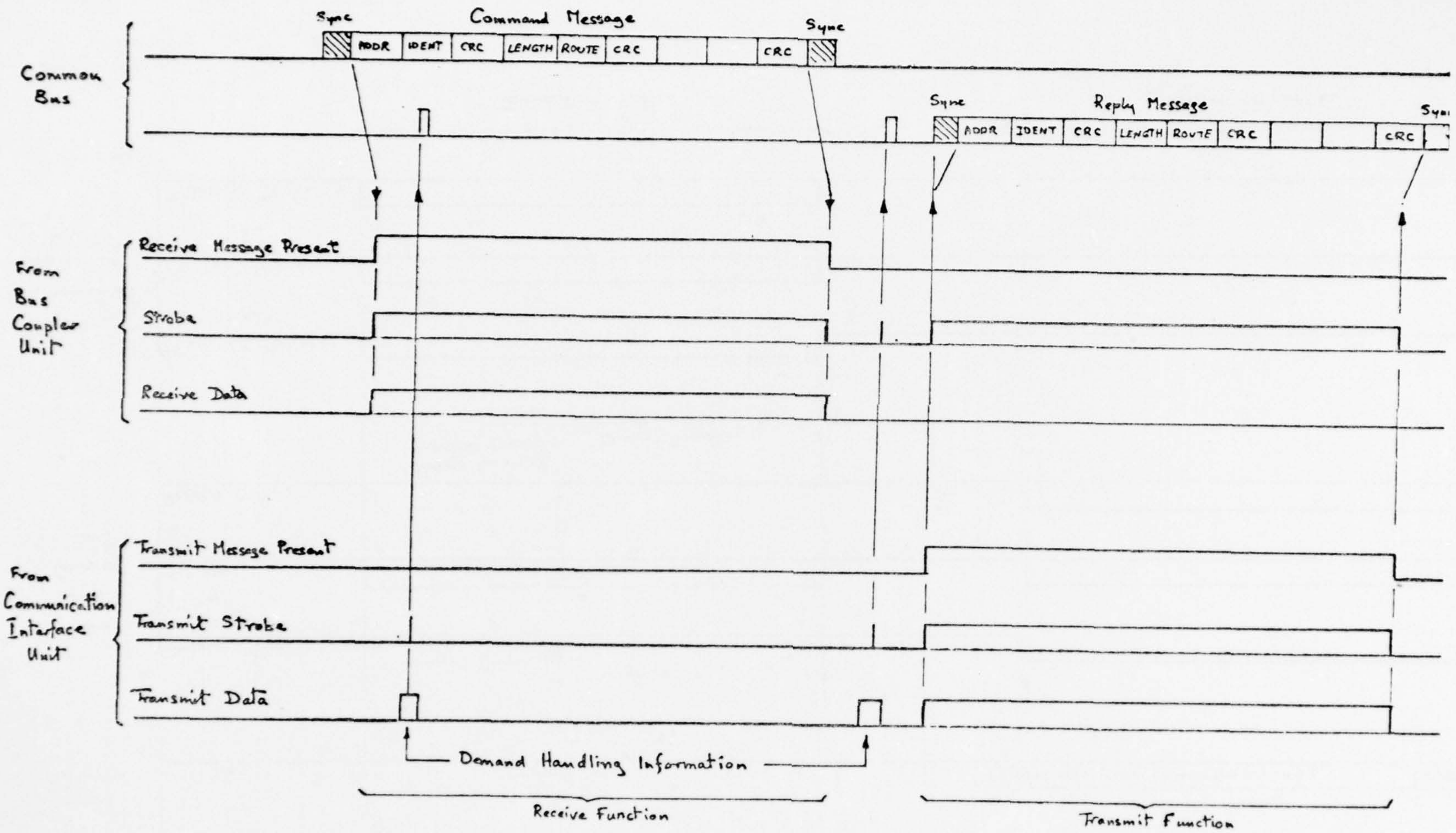


Figure 8. Example of the Usa of the Bus Coupler Port at a Slave Station

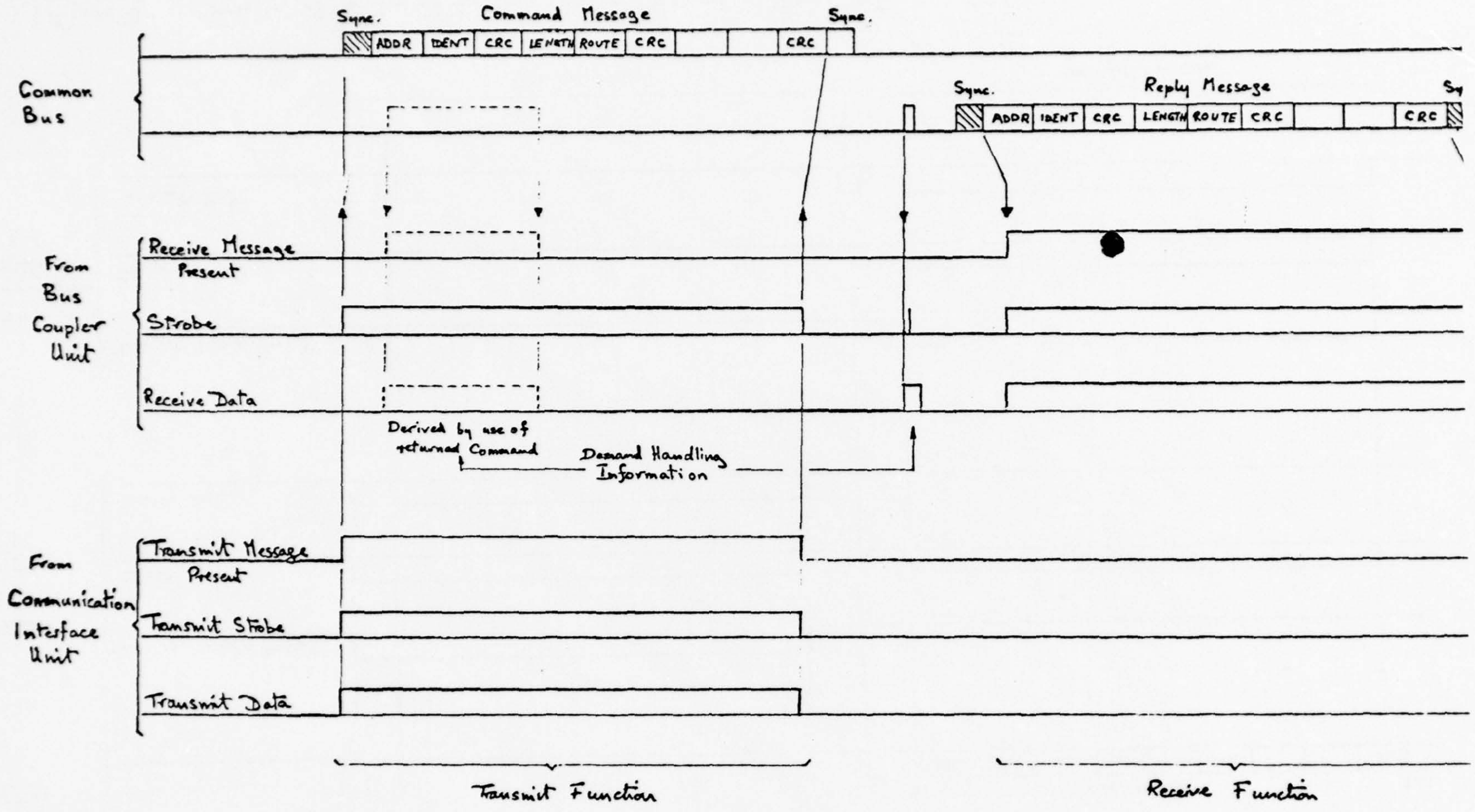


Figure 9. Example of the Use of the Bus Coupler Port at a Master Station.

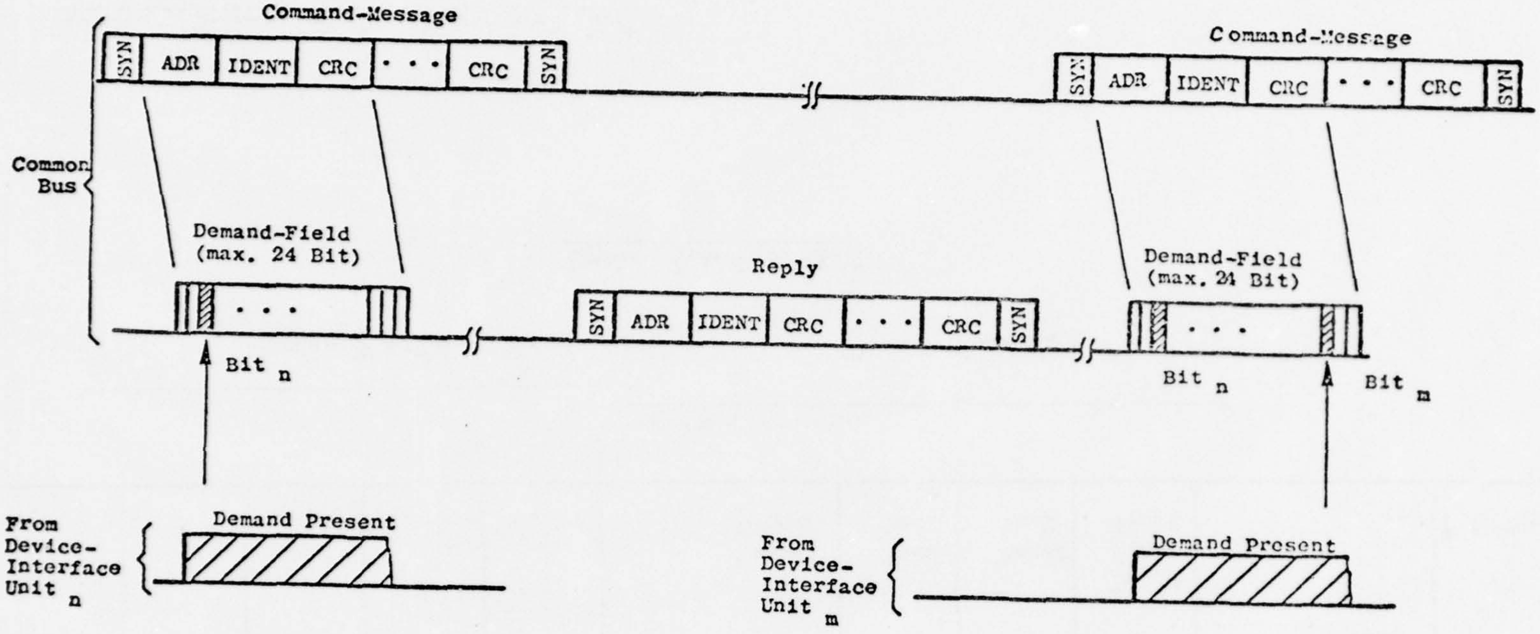


Figure 9 a Second-Level-Demand-Handling during receipt of Command Message

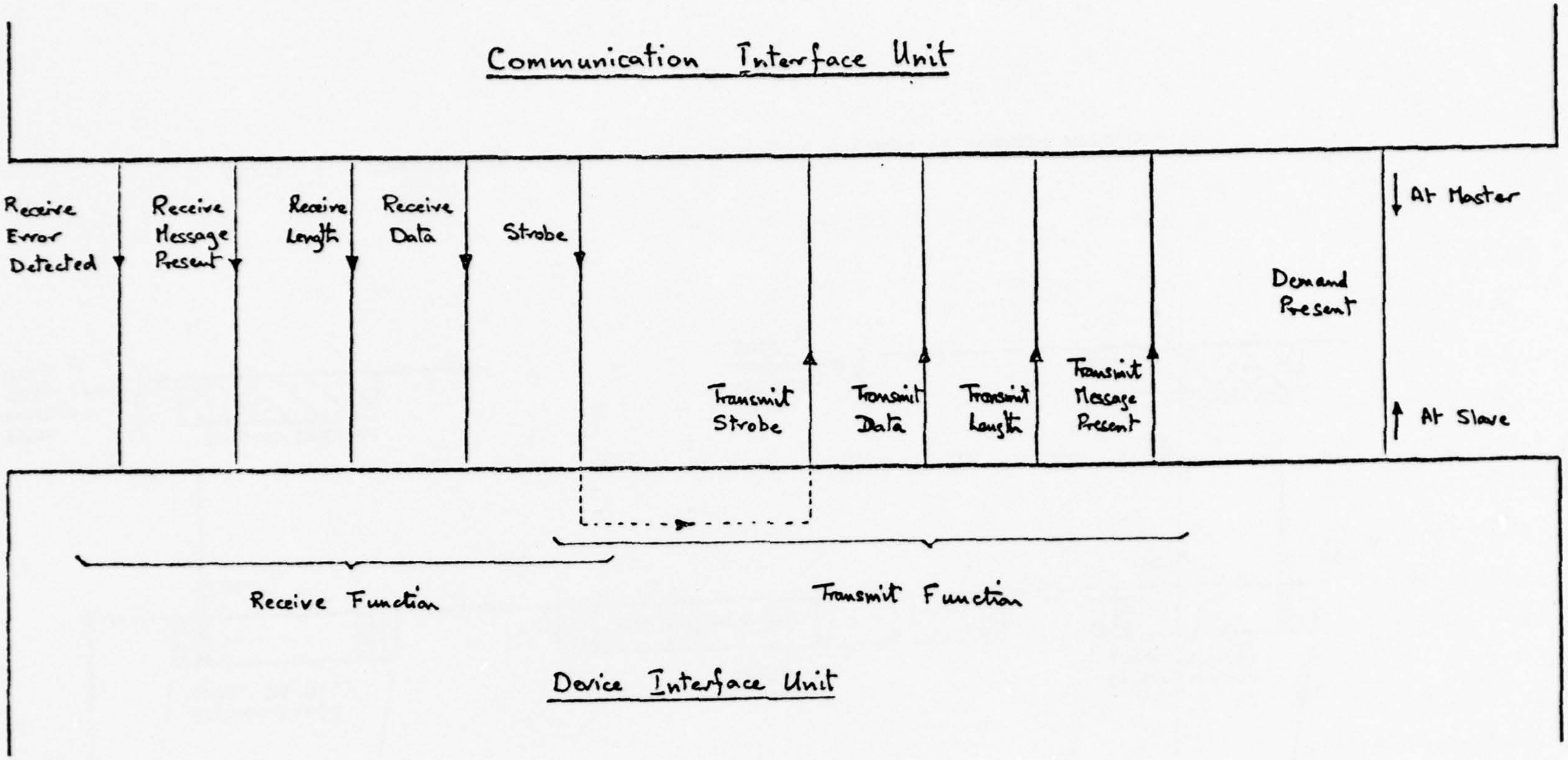


Figure 10. Signals at the Independent Port

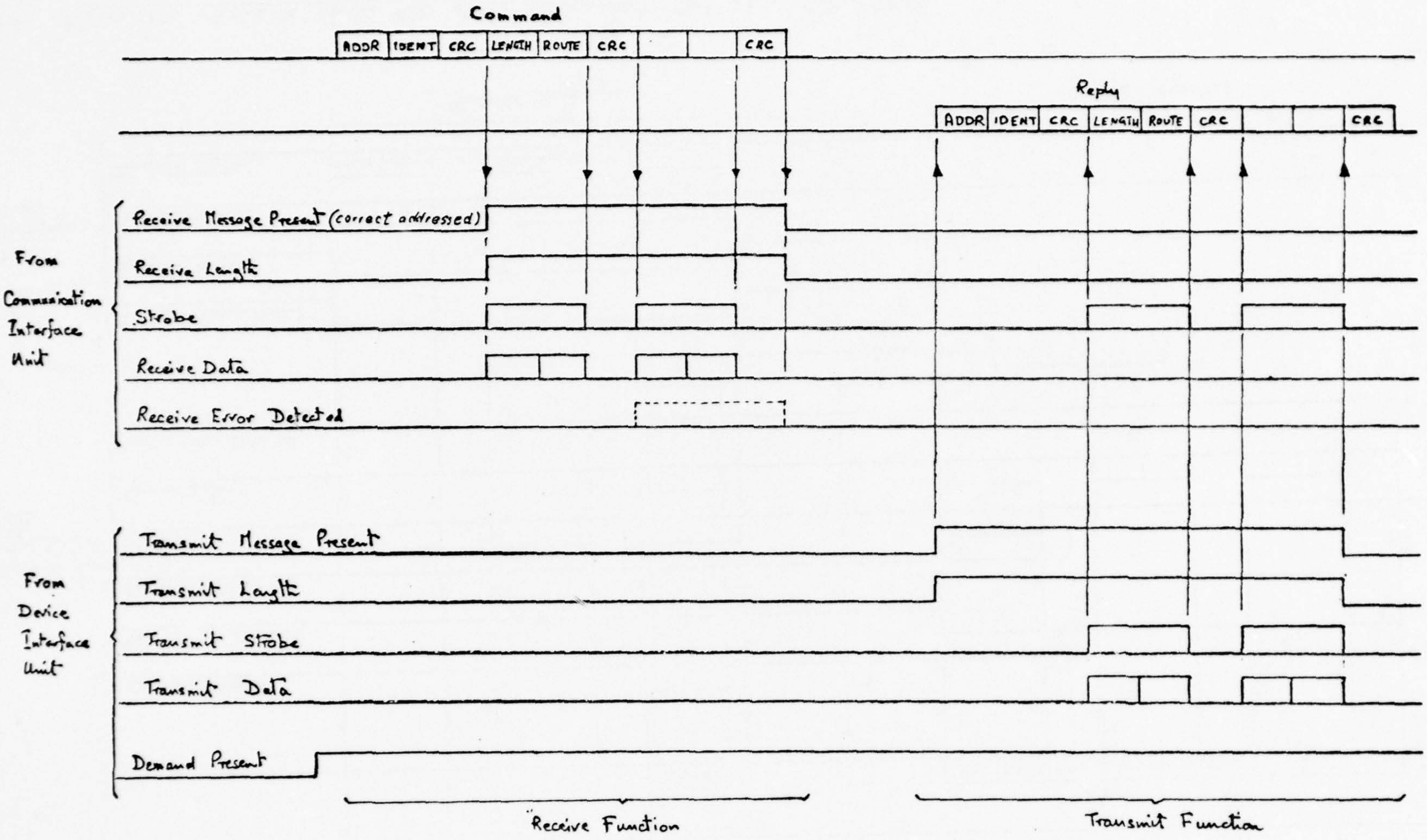


Figure 11. Example of the Use of the Independent Port at a Slave Station.

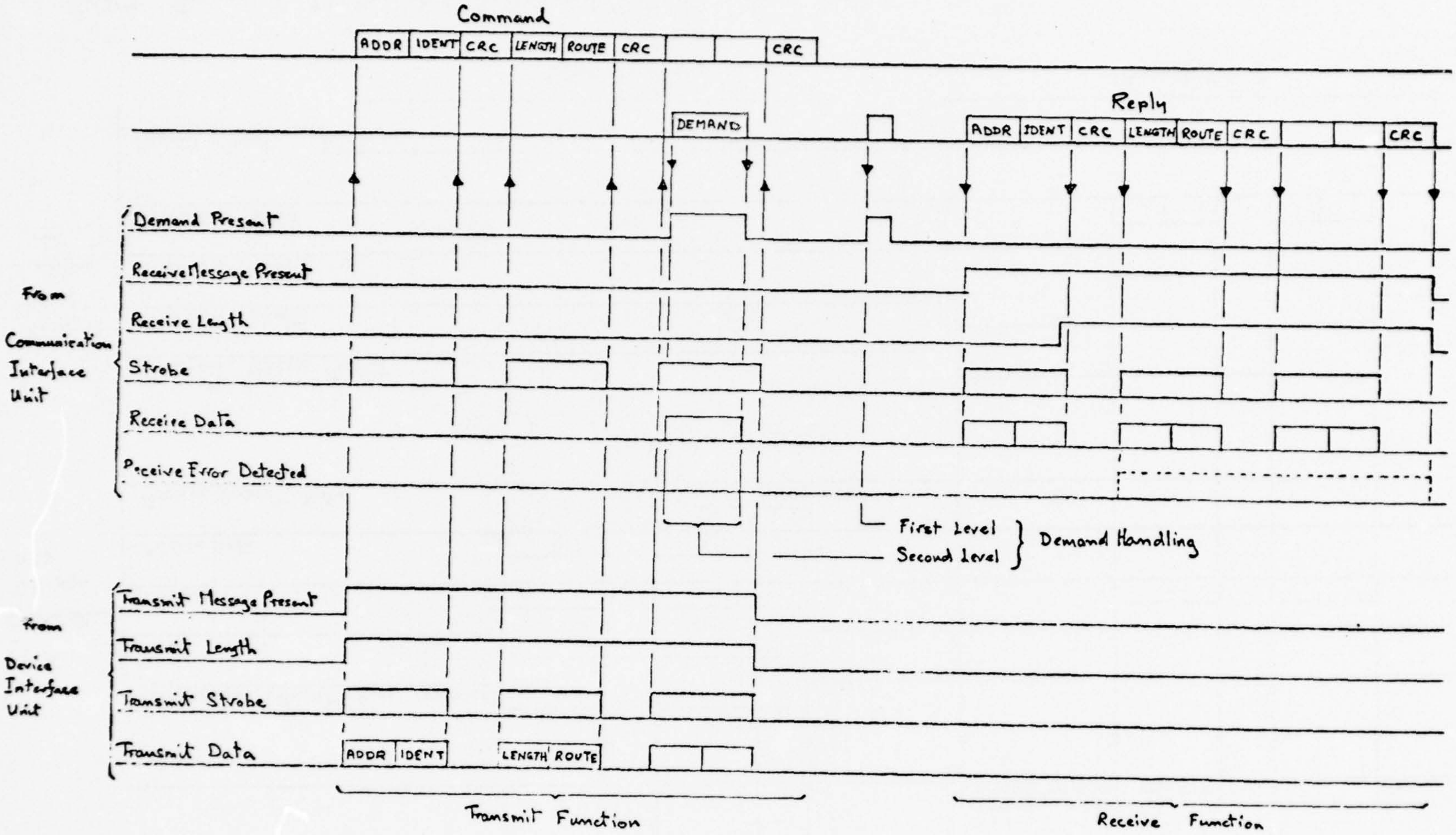
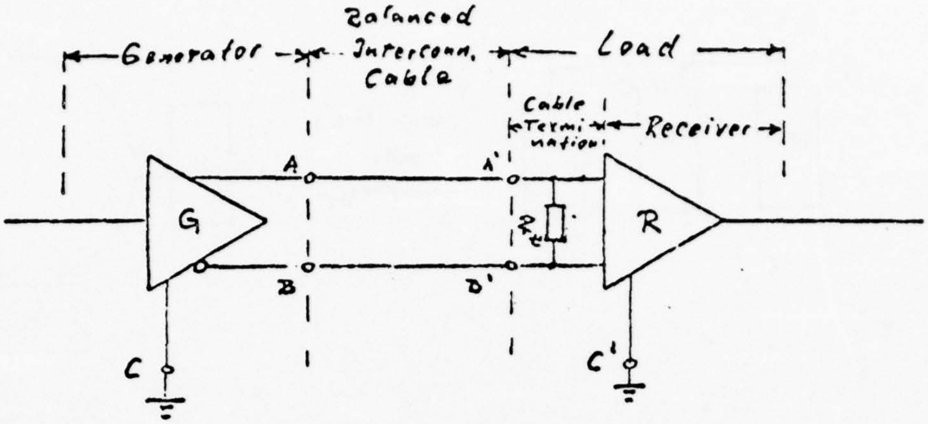


Figure 12. Example of the Use of the Independent Port at a Master Station.



R_t : Optional Cable Termination Res.

A,B : Generator Interface Points

A,B, : Load Interface Points

C : Generator Circuit Ground

C' : Load Circuit Ground

Modul. Rate / Bands	Cable length / Feet
10 K	4000
100 K	3500
1 M	350
10 M	40

Voltage Range : 2 . . . . . 6 Volts

"1" : The A Terminal of G is negative with respect to B

"0" : The A Terminal of G is positive with respect to B

FIGURE 13

EIA STANDARD RS 422

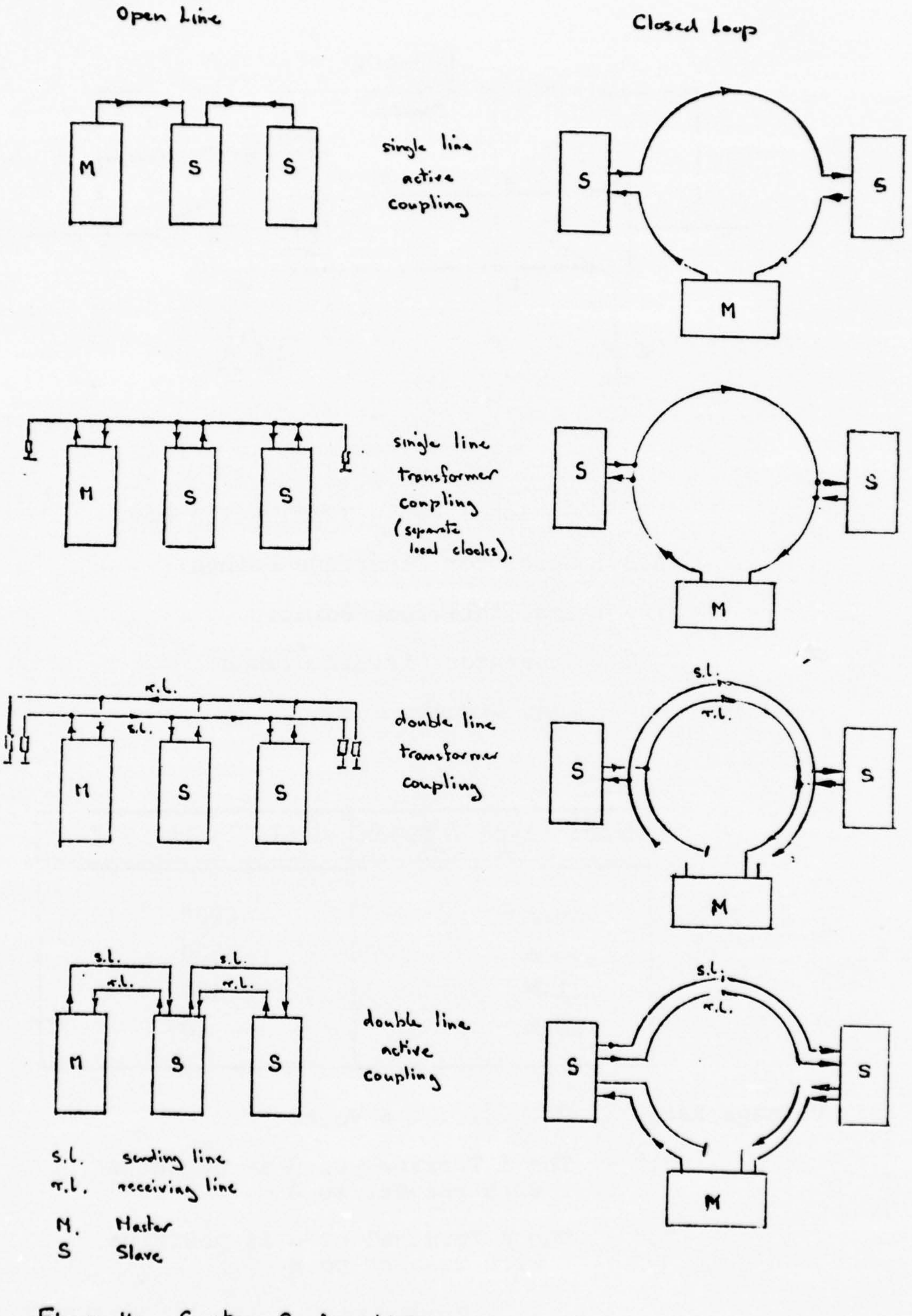


Figure 14. System Configurations.

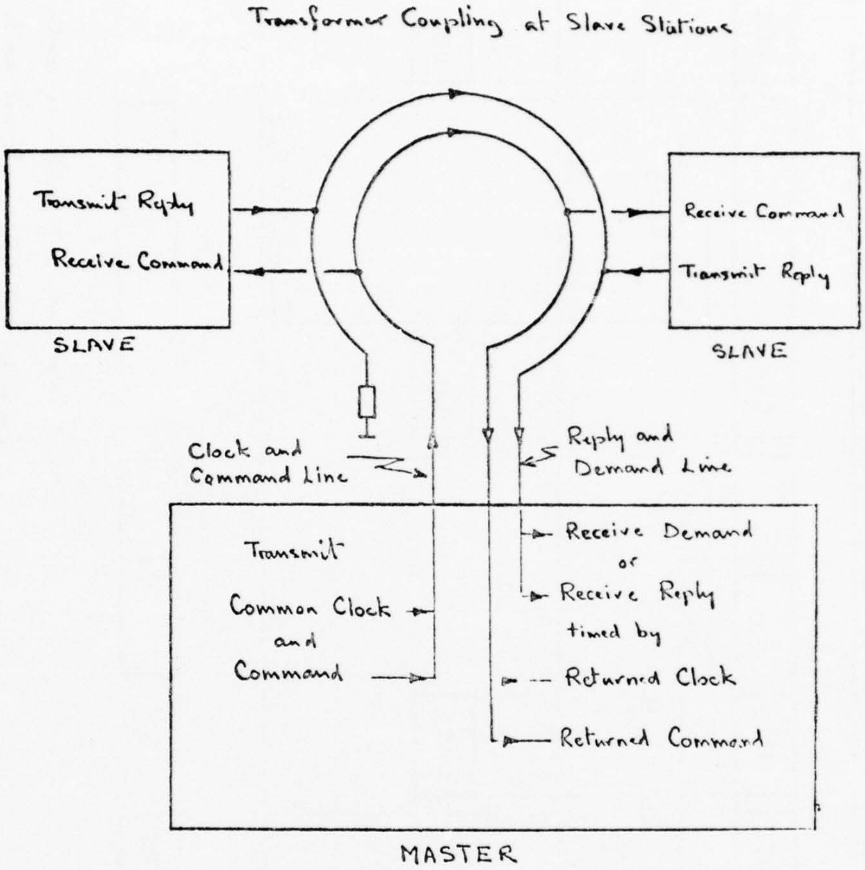
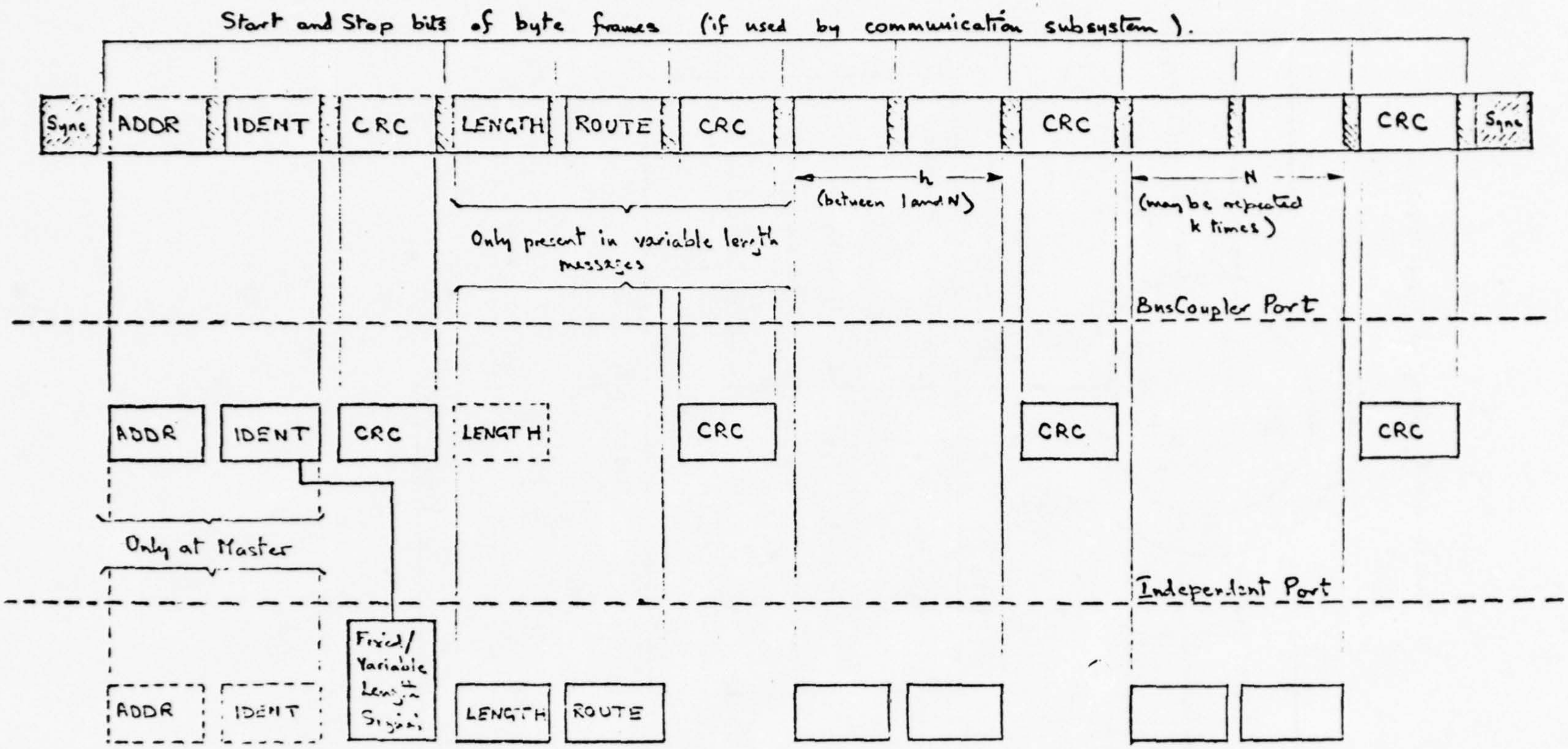


Figure 15. A Preferred Configuration.


Start and Stop bits of byte frames (if used by communication subsystem).
Sync
ADDR
IDENT
CRC
LENGTH
ROUTE
CRC
CRC
CRC
Sync
h
(between 1 and N)
N
(may be repeated k times)
Only present in variable length messages
BusCoupler Port
ADDR
IDENT
CRC
LENGTH
CRC
CRC
CRC
Only at Master
Independent Port
ADDR
IDENT
Fixed/ Variable Length Signal
LENGTH
ROUTE

Device Dependent Information is contained in $(h + kN) \leqslant 256$ 8-bit-units.

Figure 16. Information passing through the Defined Ports

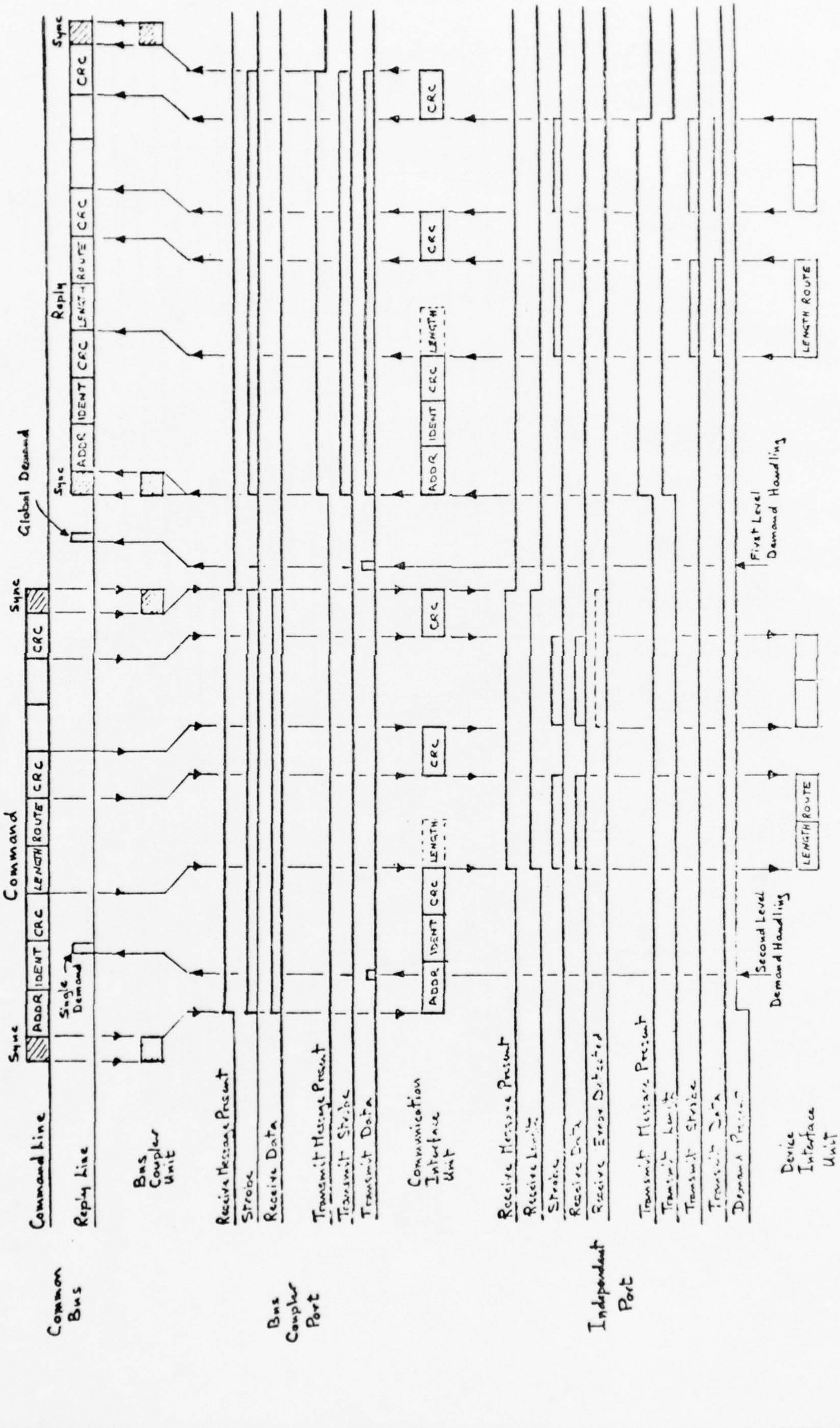


Figure 17. Use of the Ports at a Slave Station

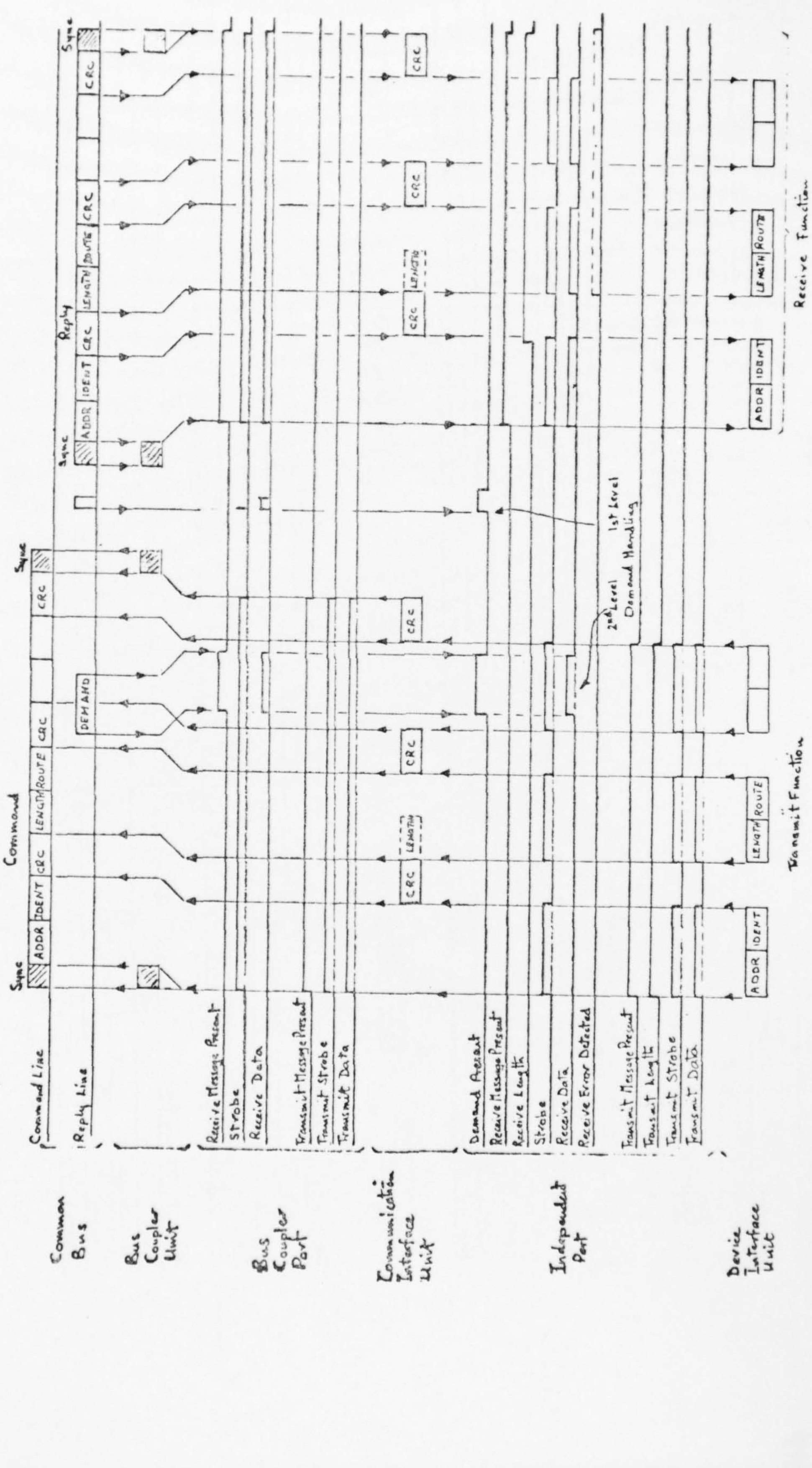


Figure 18 The Use of the Ports at the Master Station

II) Acknowledgement

The chairman of TC 5 "Interfaces and Data Transmission" thanks all active members who contributed to the draft by papers, discussions and useful criticism. Especially he thanks I.N. Hooton who put great effort in the editorial work on that paper.

IV) Active members of TC 5 who contributed to the draft

Name	Title	Company/Institute	Address
Herbert Stocker	Dipl.-Ing.	Institut für Regelungstechnik und Prozeßautomatisierung	D 7ooo Stuttgart 1 Seidenstraße 36
Chris Vissers	Ir.	Twente University of Technology	NL Dep. EL POB 217 Enschede
D. Janetzky	Dipl.-Ing.	Siemens AG, Energietechnik, Systemtechn. Entwicklung	D 75oo Karlsruhe Rheinbrückenstr. 5o Postfach 211o8o
I.N. Hooton		C.S.S. Division A.E.R.E. Harwell Didcot, Oxford	GB Oxford England
K. Müller	Dr.	Kernforschungsanlage Jülich, ZEL-NE	D 517o Jülich Postfach
Günther Haussmann	Dipl.-Ing.	AEG-Telefunken	D 775o Konstanz Bücklestraße
Peter Mielentz	Dipl.-Ing.	Brown, Boveri & Cie.	D 68oo Mannheim Kallstadter Str. 1
K. Zenner	Dipl.-Ing.	Werkzeugmaschinenlabor der Techn. Hochschule	D 51oo Aachen Wüllnerstr. 5
R. Möhl	Dipl.-Ing.	Werkzeugmaschinenlabor der Techn. Hochschule	D 51oo Aachen Wüllnerstr. 5

Name	Title	Company/Institute	Address
Tamas Boromisza	MSEE	MMG - Automation Works Institut for Research and Development	H 13oo Budapest P.O. Box 59
Manfred Mall	Dr.-Ing.	Dornier System GmbH	D 799o Friedrichshafen Postfach 648
H. Welfonder	Dr.	Institut für Verfahrenstechnik und Dampfkesselwesen der Universität	D 7ooo Stuttgart 8o Pfaffenwaldring 23
J. Biri		Central Research Institute for Physics KFK I	H 1525 Budapest POB 49
W. Attwenger	Dr.-Ing.	Österr. Studiengesellschaft für Atomenergie Electronics-Dept.	A 1o32 Wien Lenaugasse
Klaus Zwoll	Dr.-Ing.	Zentrallab. Elektronik, Kernforschungsanlage Jülich	D 517o Jülich Postfach 1913
Graeme Wood		Foxboro-Yoxall Ltd.	GB Redhill, Surrey England RH1 2HL
S. Keresztely	Dipl.-Ing.	Hungarian Academy of Sciences, Research Institut for Automation and Computing	H 1111 Budapest XI Kende 12-17
H. Walze (Chairman)	Dipl.-Ing.	Gesellschaft für Kernforschung mbH, Projekt PDV	D 75oo Karlsruhe Postfach 364o

REQUIREMENTS FOR
ONSITE REMOTE MULTIPLEXING

1. Reliability equivalent to single hard wired or pneumatic tube loop.

2. Equipment modular construction so expansion uses same transmission wires.

3. Online maintenance and calibration.

4. Intrinsically safe.

5. Signal system compatible with computer and instruments in control center.

6. Field units in all-weather housings.

7. Transmission systems unaffected by outside radio and electrical interference.

8. Field multiplexer have signal and power I/O isolation.

9. Scan speed per point that is adequate for fast response loops, and allow expanding of multiplexer to full capacity and keep same scan speed.

10. Handle digital and analog signals on a random mixed basis.

11. System accuracy ≈ ± 0.1% error.

ONSITE (SAMPLE TIMES)

For DDC Control Loops (i.e., inputs directly associated with simple and cascade loops)

	Seconds
Flow Loops	2
Pressure Loops	4
Level Loops (Holdup ≤ 3 min.)	4
Level Loops (Holdup > 3 min.)	8
Fast Acting Temperature Loops (Liquid Mixing)	8
Temperature Loops	16
Analyzer Loops	16
Valve Position Controllers	16

For Supervisory Inputs (inputs used for supervisory programs, flow integrations, material balance, etc.)

All Flow Inputs	8
All Other Inputs	16

Typical Loop Distribution

	Control Loops		Scan	Class
50%	Flow Loops	250	2 Sec.	125.0 Pts/Sec.
20	Pressure Loops	100	4	25.0
5	Low Holdup Level Loops	25	4	6.2
5	High Holdup Level Loops	25	8	3.1
5	Fast Temp. Loops	25	8	3.1
15	Slow Temp. Loops	75	16	4.7
		500		167.1
	Additional Inputs (for supervisory calculations)			
50%	Flow Inputs	350	8	43.8
50	Other Inputs	350	16	21.9
				65.7
				232.8 Pts/Sec.

TYPICAL DISTRIBUTION

FOR SCAN ACTIVITY BY ELEMENT (OFFSITES)

Element	1 sec	10 sec	1 min	3 min	5 min	10 min	1 hr
Tank Levels							
● 4-20mA				7		7	26
Valves							
● ROV	10		466				
● MOV Position			15				
Pumps	1		37				
Mixers	1				53		
Temperatures							
● Tanks						151	
● PDM			64				
PDM/Turbine			64				
Analog 4-20mA (pH, Flow)			7				
Weighbridge/Badge Reader							
● Data Ready	5						
High Level Alarms (HLA)		46					

INDEPENDENT INTERFACES

These figures show how various levels of hardware and software can be used to achieve mutually independent interfaces. The computer, purported to be a truly general-purpose device, lies at the center, on the dotted line, with its all-purpose executive system. For a given process application, transducers and actuators are needed that are tailored to that process, as shown by the bottom layer of the structure. Each process application also has its own software, to implement the desired control strategy (top layer). These layers are related to each other, and to the process, but not to the computer itself.

All levels inside these outer layers of the structure can be independent of the process.

The transducers (and actuators) are driven by standardized I/O modules, which are programmed by standard drivers. Each I/O module has its own driver.

In the center of the structure, a specific interface (hardware) module is used to transform the I/O bus of a given computer into a standard I/O bus. A different module is probably required for each different computer, but only one such module is required for each computer, if only one I/O bus standard is used. This specific interface requires one specific software driver for that particular computer, to interface with the standard module drivers.

The number of specific driver/specific interface combinations required to make N computers interchangeable with M different I/O bus combinations is the product N*M. Similar logic applies to the variety of I/O equipment and transducers required for various interfaces.

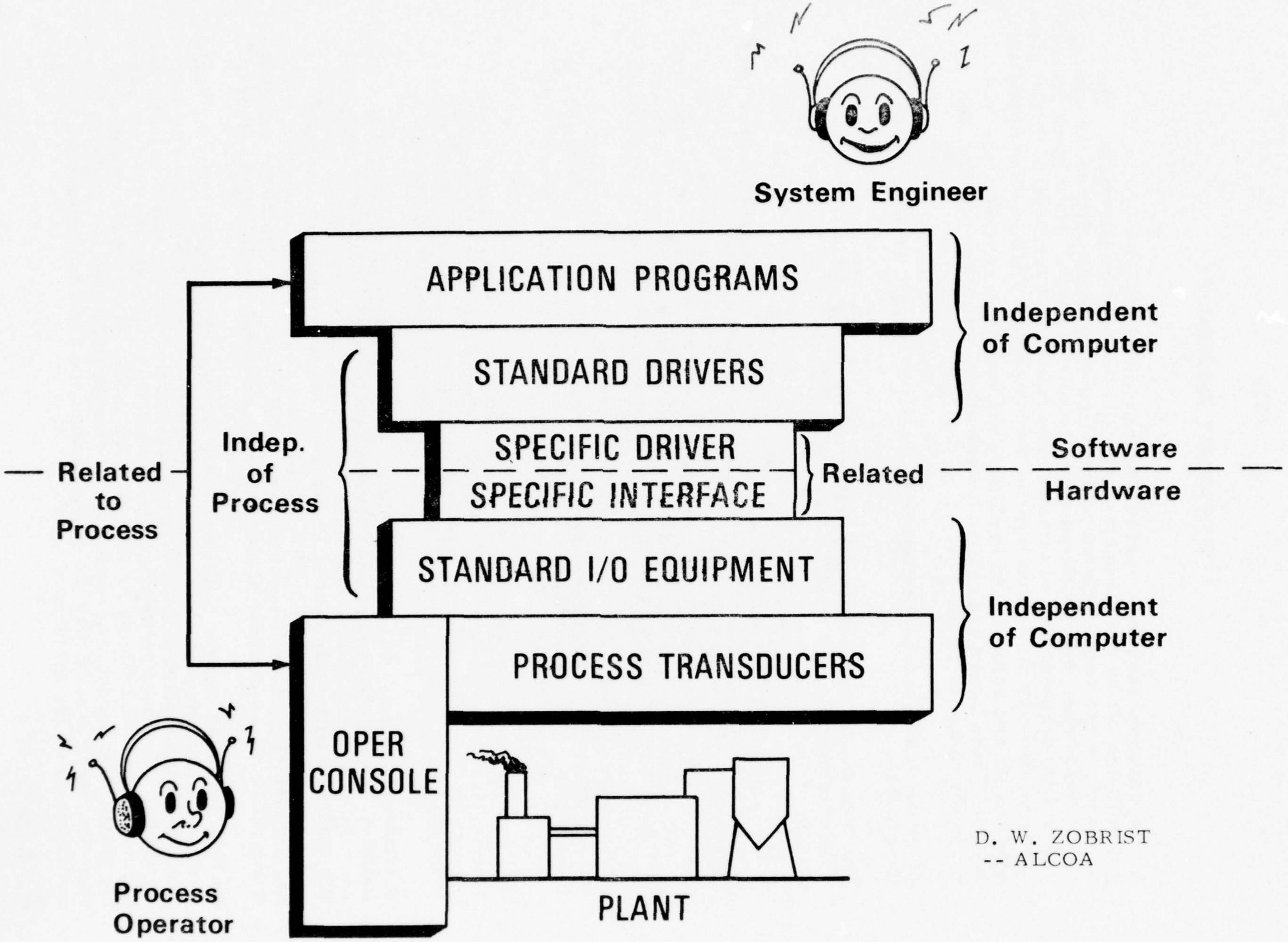
The CAMAC standard offers the promise of having one I/O bus that could interface with all I/O equipment, holding the numbers of combinations to a minimum. It standardizes the bus between the specific interface and the standard I/O equipment, and makes the standard I/O equipment possible.

A structure such as this is needed to stabilize the process control computer industry to the extent required to develop second sources (and complementary sources) of equipment and computers for control purposes, and to achieve a life cycle for such equipment of 15 years (as is expected of non-computer control equipment in industry). Such stabilization can properly apply to mechanical and electrical interchangeability, and still allow for competition and technological progress in areas of cost, speed, functional capability and others.

The figure can also be used to illustrate the fact that the process operator tends to view his plant through the operator's console, while the systems engineer tends to view the plant (and the control system) through the entire engineered structure. The more transparent this structure is, that is, the less effort the engineer spends in building it, the better he is able to direct his attention to the plant. The headphones represent the thought that, until the engineer can view the plant the same way the operator does, he and the operator had best communicate on the same wave length.

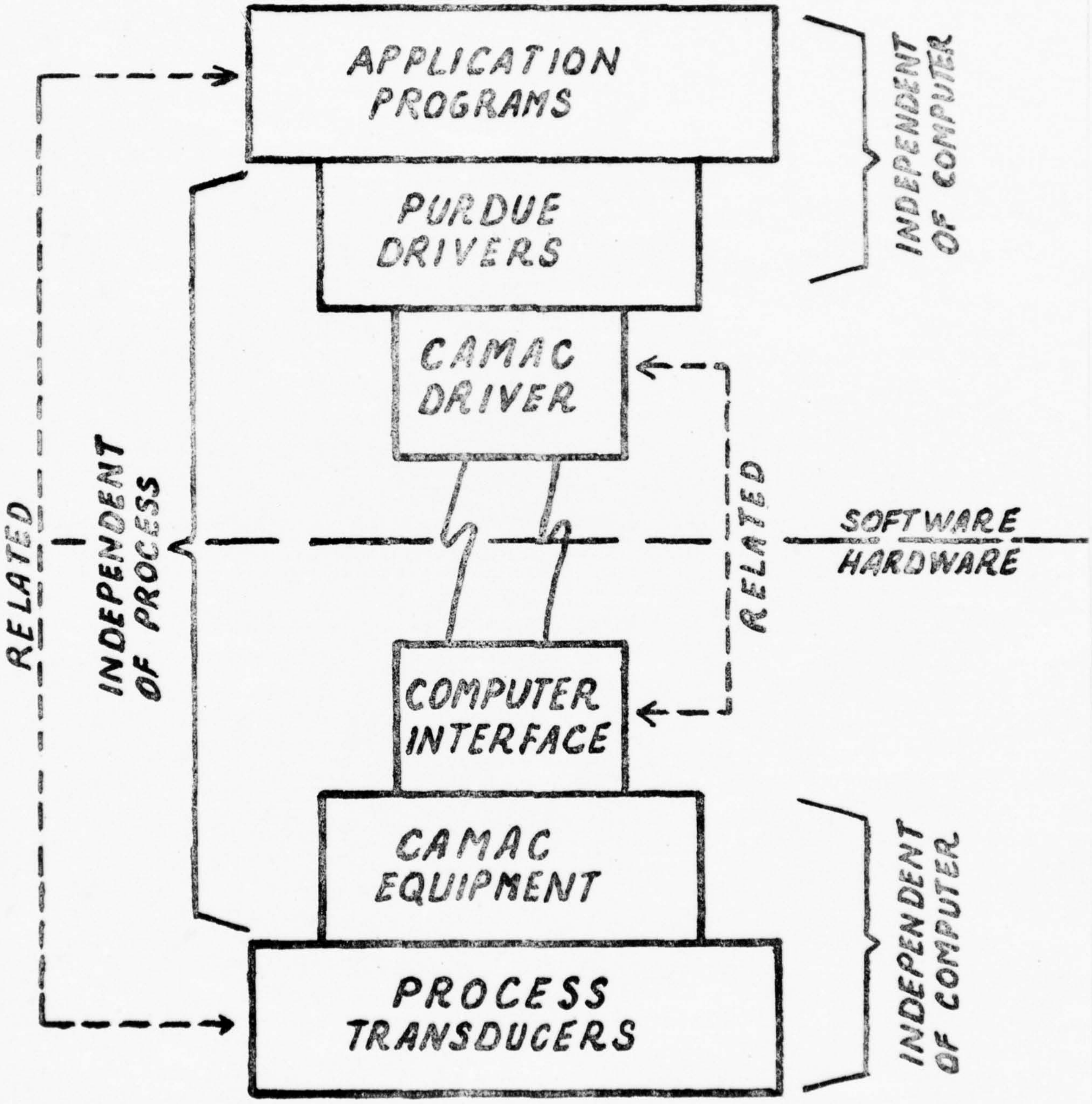
Sincerely,

Dick
R. L. Curtis



D. W. ZOBRIST
-- ALCOA

Using Isolation to Achieve Independent Interfaces.


APPLICATION PROGRAMS
PURDUE DRIVERS
CAMAC DRIVER
COMPUTER INTERFACE
CAMAC EQUIPMENT
PROCESS TRANSDUCERS
INDEPENDENT OF COMPUTER
INDEPENDENT OF PROCESS
RELATED
RELATED
SOFTWARE
HARDWARE
INDEPENDENT OF COMPUTER

ALUMINUM COMPANY OF AMERICA
ALCOA BUILDING · PITTSBURGH, PENNSYLVANIA 15219

ALCOA

(412) 553-2199

February 25, 1974

Dr. T. J. Williams
PLAIC
Purdue University
West Lafayette, IN 47907

Mr. Paul H. Berka
Aluminum Company of America
Alcoa Technical Center
Alcoa Center, PA 15069

Dear Ted and Paul:

Re: Implementing CAMAC Serial Highways

You have both been interested in methods for implementing the CAMAC serial data highway and in providing suitable redundancy in the data paths to increase the total system reliability. The attached sketches show some of my thoughts on the subject. Most of the techniques shown can also be used with serial highways other than CAMAC. Other methods can also be used to provide the desired features.

Type L-1 Serial Crate Controllers

The type L-1 SCC will most likely be used in nearly all future CAMAC serial highway systems. The economics of using mass-produced units, and adding an external box for any additional functions which may be required for a particular installation, will no doubt be more favorable than custom-designed crate controllers.

The L-1 SCC has had considerable engineering applied to its design. It includes compromises between maximum capability and minimum requirements so that it should be useful in a very broad range of applications. I think it is a reasonable design to optimize the standard unit.

The clock and data signals are separated for two reasons: (1) for use in the byte-serial mode, and (2) to keep the costs down. It permits interconnecting more than one crate at one location into a crate cluster using two pairs of twisted wires. This is less expensive than including modulators and demodulators each time to combine and separate the data and clock signals.

To go long distances it may be cheaper to put in modems and transmit the combined signals over a single circuit. Various techniques are available for this: frequency modulation, phase modulation, pulse-width modulation, etc. I will discuss merits of different clocking schemes another time.

The L-1 SCC provides the necessary control logic for bypassing the crate when it is off-line, and for additional programmed loop-path control (e.g., loop collapse). The L-1 does not include the actual switching of the loop signals. This is the best, I think, since different applications and installations will most likely use different loop circuits (balanced twisted pair, unbalanced coax, fiber optics, telephone lines, etc.). Many systems may not require any loop switching at all.

The data and clock signals (the minimum signals required to operate a CAMAC serial highway) go in and out of the L-1 as balanced twisted pairs. This permits very low-cost implementations of the highway where the distances are not great. To go any significant distance (e.g., hundreds of feet) I think our preference will be to use unbalanced coaxial cables carrying combined clock and data signals. However, we also have shorter distance requirements, such as across the room, crate clusters, etc. I'm certain we will also find instances where telephone lines are useful.

Crate Bypass

When a crate is taken off-line, whether intentionally or due to a power failure or other malfunction, it is often desirable for the remainder of the serial highway to continue functioning. While a crate is off-line the incoming serial highway signals must then be passed on to the next crate without alternation. Figure 1 shows a crate being bypassed. The off-line crate may continue to monitor the incoming signals (as long as it has power) to watch for "turn-on" commands from the serial driver. While the crate is off-line, however, the system does not depend upon it to monitor, amplify, or reshape the signals for the other crates.

It is expected that bypass switching will normally be implemented with electromechanical relays. This enables positive switching action to take place if power is lost at the crate, i.e., failsafe operation.

Alternate Paths

In large systems where high reliability is essential, alternate signal routes may be in order in case a cable should fail, e.g., accidentally cut. Figure 2 shows one such system using a double loop. The second cable is the alternate or backup cable. It is used when a section of the primary cable fails. Figure 3 shows one method of using the alternate cable and figure 4 shows another method.

The second method (figure 4) is probably best suited for the ship-board applications you are considering, Ted. Many of our industrial plant applications fall into a similar category, where loss of the primary cable at one location may likely be accompanied with loss of the alternate cable at the same location (e.g., damaged conduit).

Figure 5 shows one method of detecting cable failures. The method shown uses the center conductors of the primary and secondary coaxial cables for a dc security circuit. Note that this has the added advantage of monitoring the alternate cable. (Otherwise the alternate path could fail and you might not know about it until you needed it.) The switching (alternate routing) relays have sufficient coil inductance to block the high-frequency serial data signals. The data signals are coupled to the transmitter and receiver circuits through capacitors or high-frequency transformers. A similar method can be employed for twisted pair lines by using a phantom circuit. Figure 6 shows a full complement of equipment for use with an L-1.

Note, the alternate-path switching I have shown is different than the "loop-collapse" switching indicated in the CAMAC serial highway description. Loop-collapsing normally involves the deletion from the highway of all crates farther from the serial driver. I do not see much need for this in our applications. It might, however, be used to bypass a leg of the highway going to a single process of a multi-process computer system.

Lightning Protection

Many industrial applications have a requirement for lightning protection on their signal cables. The use of large, solid outer-conductor coaxial cables, such as used for CATV systems, provides some protection when the outer conductor is well grounded. Fast-acting gas-discharge lightning protectors also help. This combination seems to be sufficient for CATV systems. I suspect it will be sufficient for many of our applications as well.

High common-mode voltages and high-energy sources, such as pot rooms, present another problem. The use of fiber optics for at least a portion of each circuit length, may provide the answer.

Signal Amplification

Extremely long distances will require amplification midway. For this reason we have been looking closely at the techniques used in CATV systems. They use amplifiers every 2,000 feet or so. Power for the amplifiers is provided by 30 or

60 volts, 60 Hertz between the center and outer conductors of the coax.. Power can be sent either direction through the cable. They also can send signals in both directions in the same cable by using different carrier frequencies.

Since CATV equipment is readily available at reasonable prices, there may be instances where it will be the best answer. Both the primary and alternate-path circuits could then be sent over the same cable. In addition, closed-circuit TV signals could also share the cable.

Note, CATV is normally a multi-drop system, not a loop configuration. A different "channel" could be used for each section of the serial highway. This may quickly use up the available bandwidth of the cable. It will be most applicable to very long highways with crate clusters at only a few remote locations.

Speed and Distance Trade-Offs

There are a number of speed and distance trade-offs which should be considered for any given installation. As a general rule the lower the speed, the farther one can go without the need for amplifying repeaters. The range of CATV systems is also a function of the cable size: the larger the cable, the lower the signal loss.

If one is using transmitters that can drive a line 500 feet at 10 Megabaud (maximum speed for CAMAC data and clock together) they probably can drive a 1,000 foot line at 5 Megabaud. To go 800 feet it may be cheaper using separate cables for clock and data than to use additional equipment to combine the signals and need an amplifier midway.

I hope this discussion has provided some useful ideas for you. To my knowledge, no one else is working on these areas which are not covered by the L-1 SCC. Work at Purdue and/or Alcoa in such areas could, I think, nicely complement the work of the NIM-CAMAC working groups.

Sincerely,

Dale

Dale W. Zobrist

DWZ/bay

Attachments

cc: Mr. Louis Costrell
Mr. F. Kirsten
Mr. D. Machen
Mr. T. L. Willmott

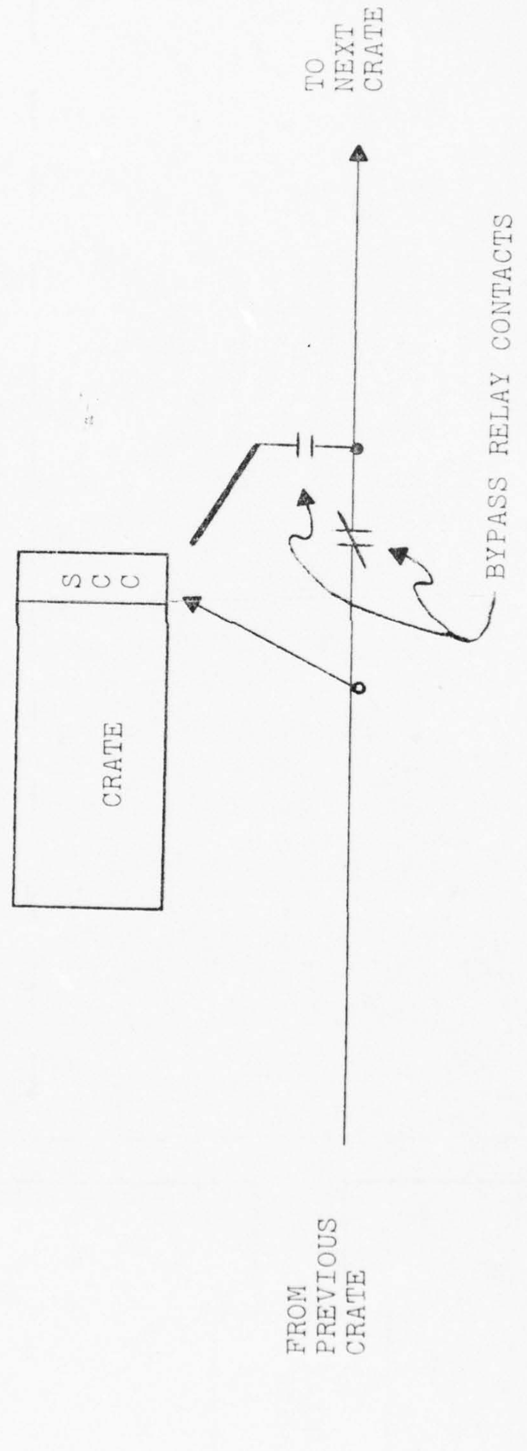

FROM PREVIOUS CRATE
CRATE
SCC
BYPASS RELAY CONTACTS
TO NEXT CRATE

FIGURE 1

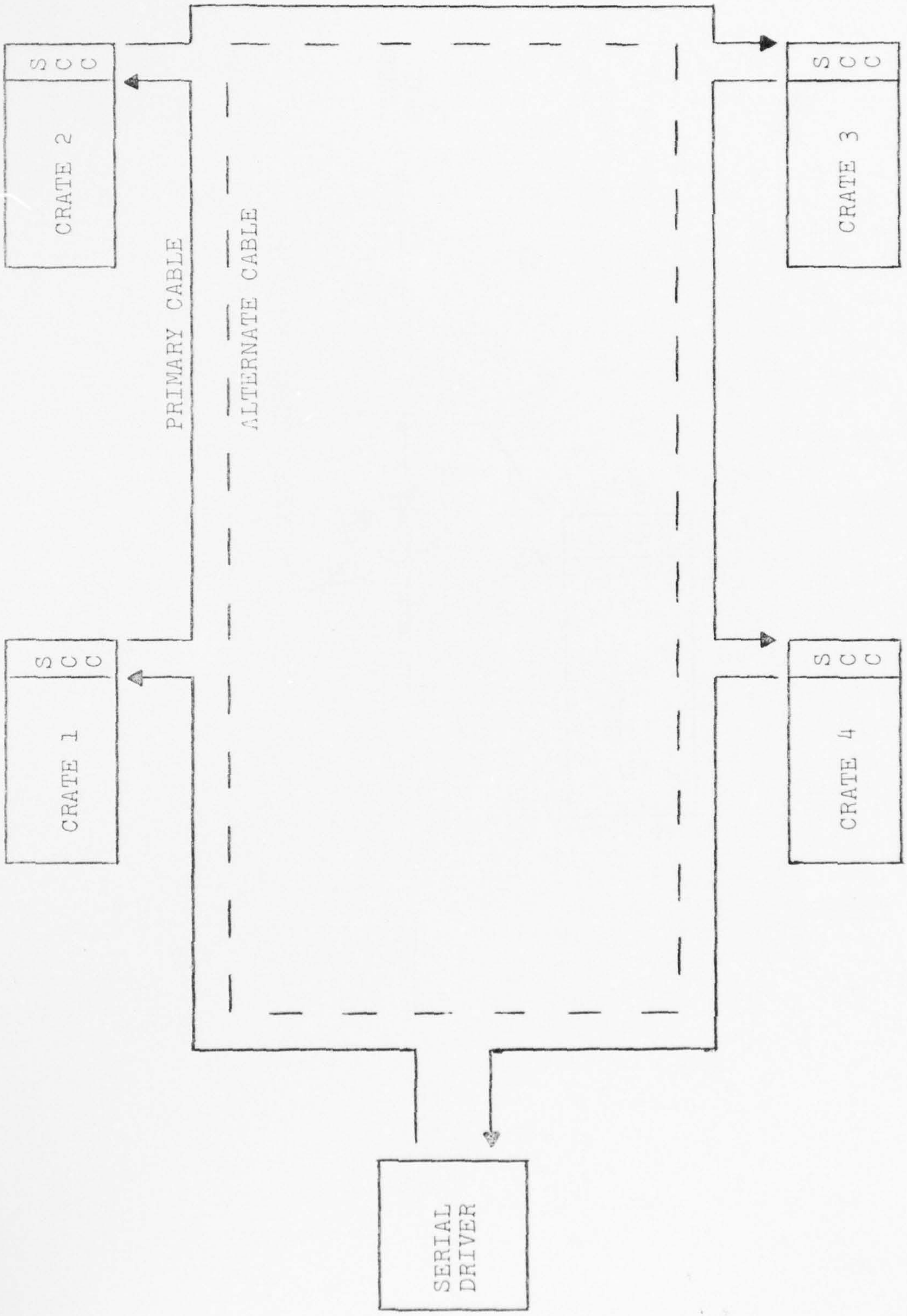

CRATE 2
SCC
CRATE 3
SCC
PRIMARY CABLE
ALTERNATE CABLE
CRATE 1
SCC
CRATE 4
SCC
SERIAL DRIVER

FIGURE 2

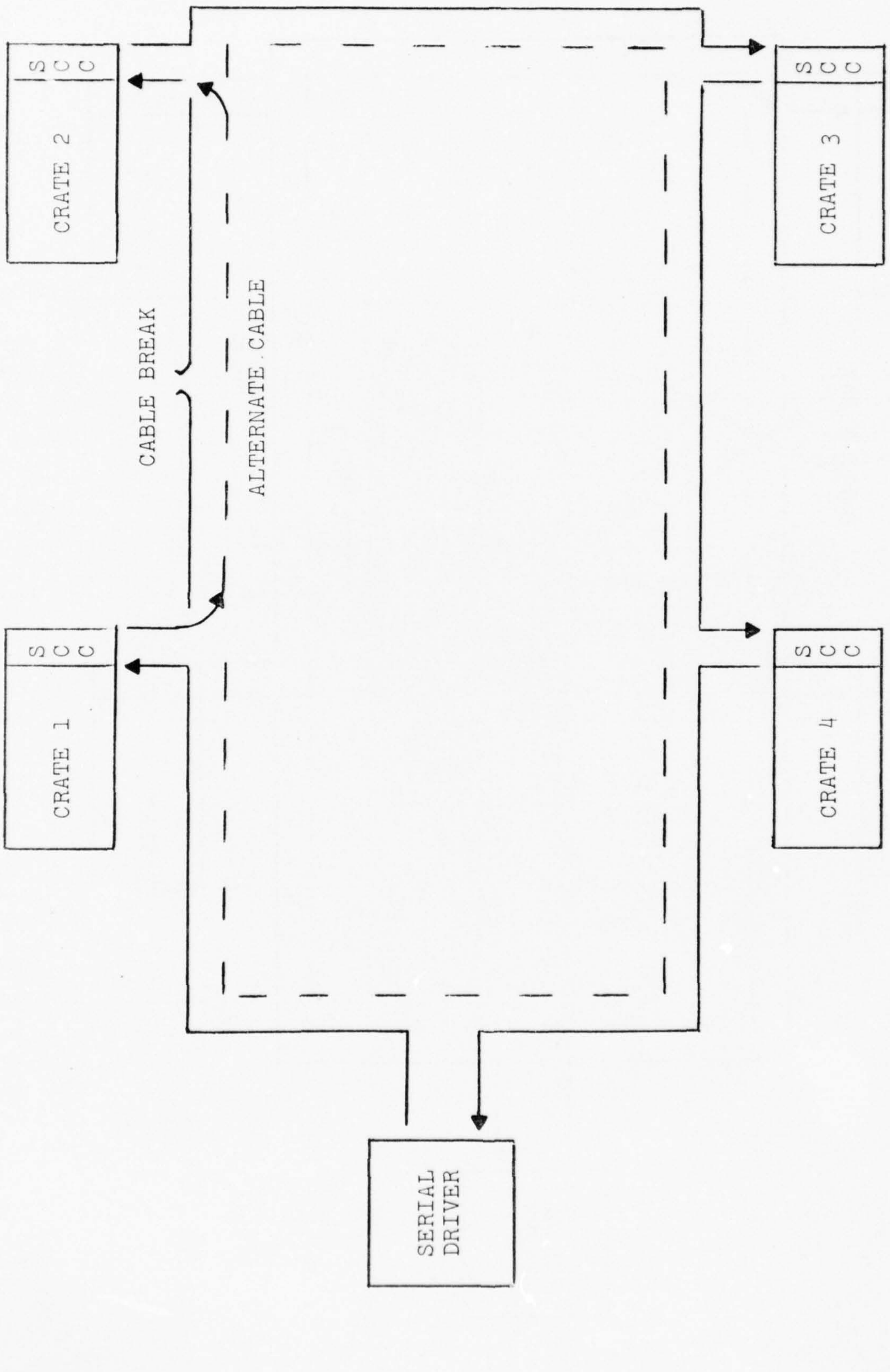

CRATE 2
SCC
CRATE 3
SCC
CABLE BREAK
ALTERNATE CABLE
CRATE 1
SCC
CRATE 4
SCC
SERIAL DRIVER

FIGURE 3

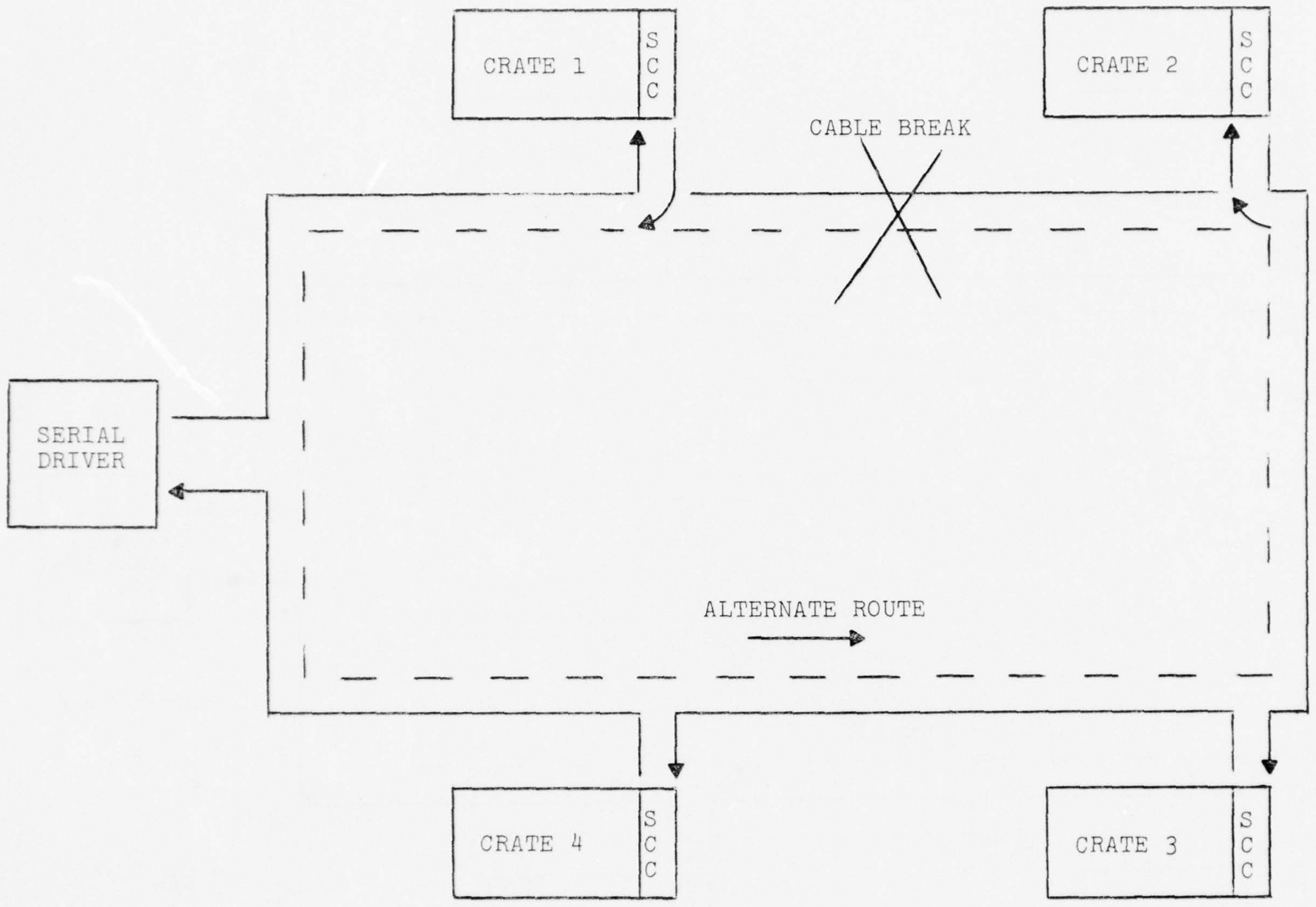

CRATE 1
SCC
CRATE 2
SCC
CABLE BREAK
SERIAL DRIVER
ALTERNATE ROUTE
CRATE 4
SCC
CRATE 3
SCC

FIGURE 4

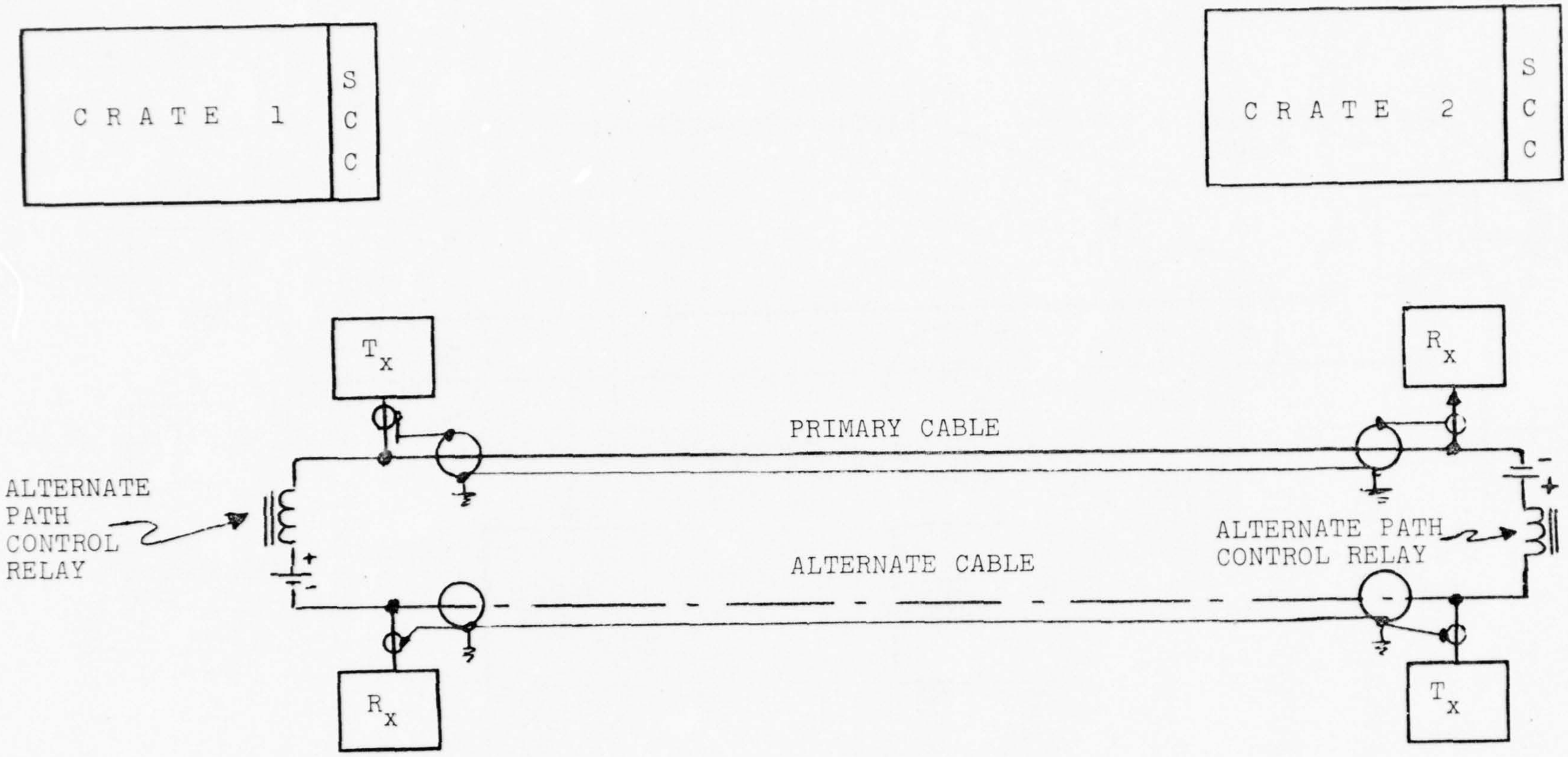

CRATE 1
S C C
CRATE 2
S C C
T_x
R_x
PRIMARY CABLE
ALTERNATE PATH CONTROL RELAY
ALTERNATE CABLE
ALTERNATE PATH CONTROL RELAY
R_x
T_x
+
-

FIGURE 5

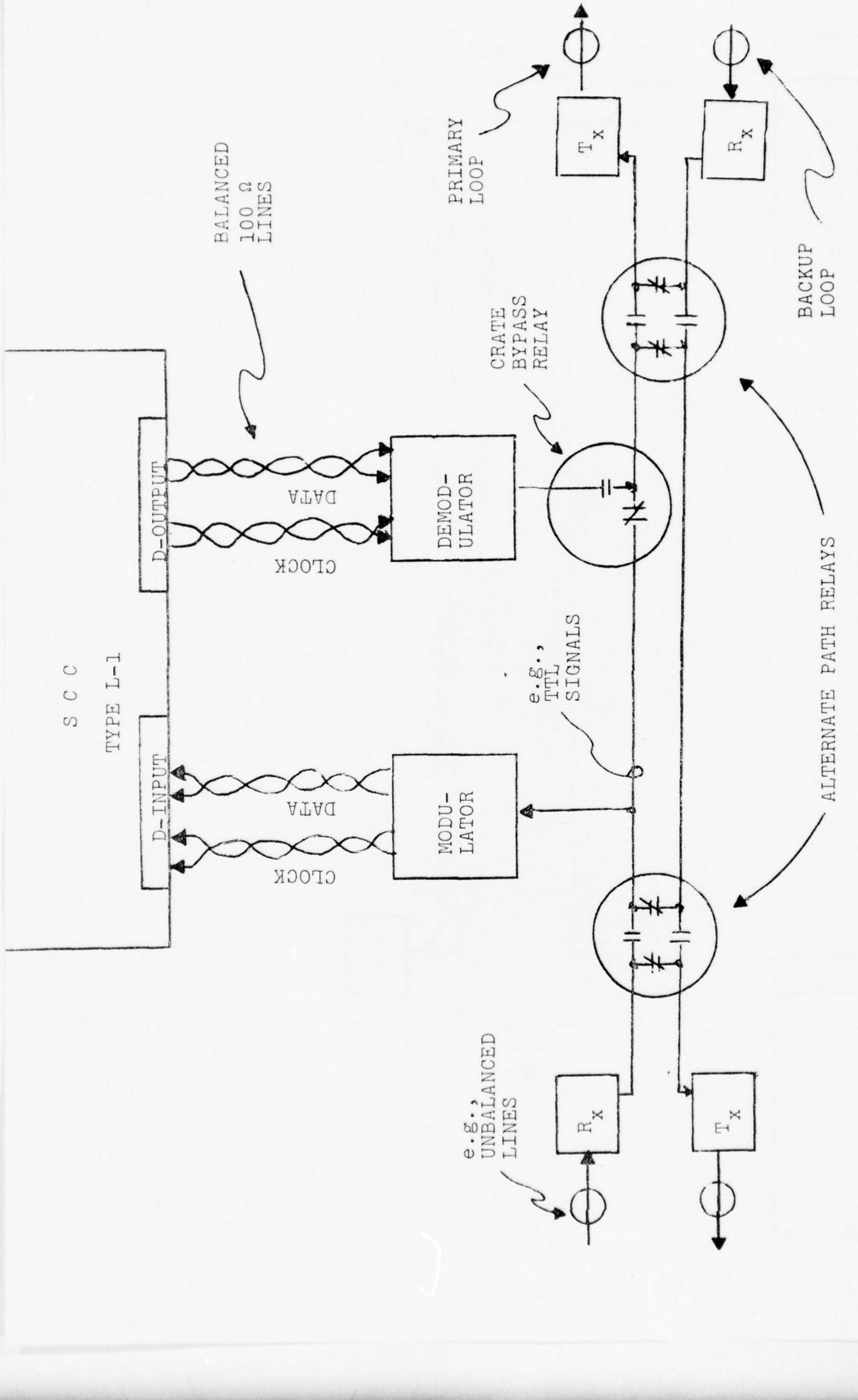

S C C
TYPE L-1
D-INPUT
D-OUTPUT
CLOCK
DATA
DATA
CLOCK
BALANCED
100 Ω
LINES
MODU-
LATOR
DEMOD-
ULATOR
e.g.,
TTL
SIGNALS
CRATE
BYPASS
RELAY
e.g.,
UNBALANCED
LINES
R_x
T_x
T_x
R_x
PRIMARY
LOOP
BACKUP
LOOP
ALTERNATE PATH RELAYS

FIGURE 6

A COMPARISON OF DATA RATE CAPABILITIES OF VARIOUS INTERFACE TECHNIQUES VERSUS REQUIREMENT OF SELECTED PROCESSES AND LEVELS OF CONTROL IMPLEMENTATION

The attached figures present material developed by the Interface and Data Transmission Guidelines Committee on the data transmission needs for the control of several representative processes, those for inter-control system communications, and the capabilities of several techniques available today.

Figure 1 describes typical regions for industrial applications. For example, fluid stream processes have distance requirement ranging from 0.1 to 1000 bits/second. Numerical control applications fall to the right of fluid processes, since the data rate requirement is slightly higher.

Figure 2 represents the regions covered by existing standards or products. Examples include 20 ma loops, which cov-r a region up to 100 bits/second, and up to 1000 feet, or the HP ASCII ubs, up to 10^6 bits/second, and up to 50 feet.

Figure 3 shows typical technology regions ranging from inter-CPU communications at high speeds and short distances, to human/machine communications at lower rates and generally longer distances.

These diagrams can be overlaid to illustrate applicability of solutions to problems. Figure 4 is an overlay of Figure 2 on Figure 1. For instance, the inference can be drawn that 4-20 ma covers only part of the fluid process applications, and none of numerical control. It is also noteworthy that CAMAC (if the diagram were to be interpreted literally) is the only standard shown for medium distance applications.

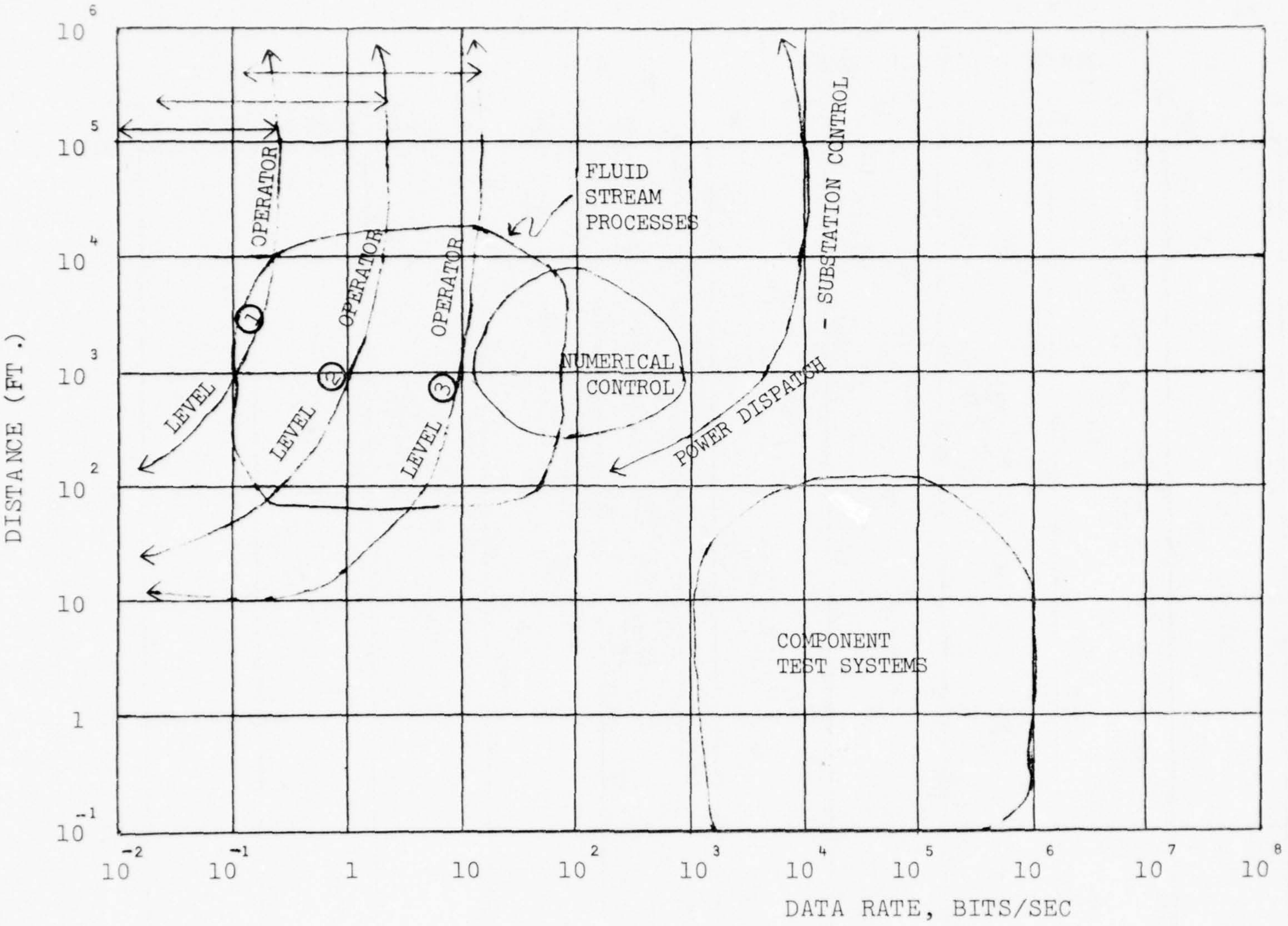


FIGURE 1

DATA TRANSMISSION REQUIREMENTS FOR PROCESS CONTROL OF SEVERAL REPRESENTATIVE PROCESSES

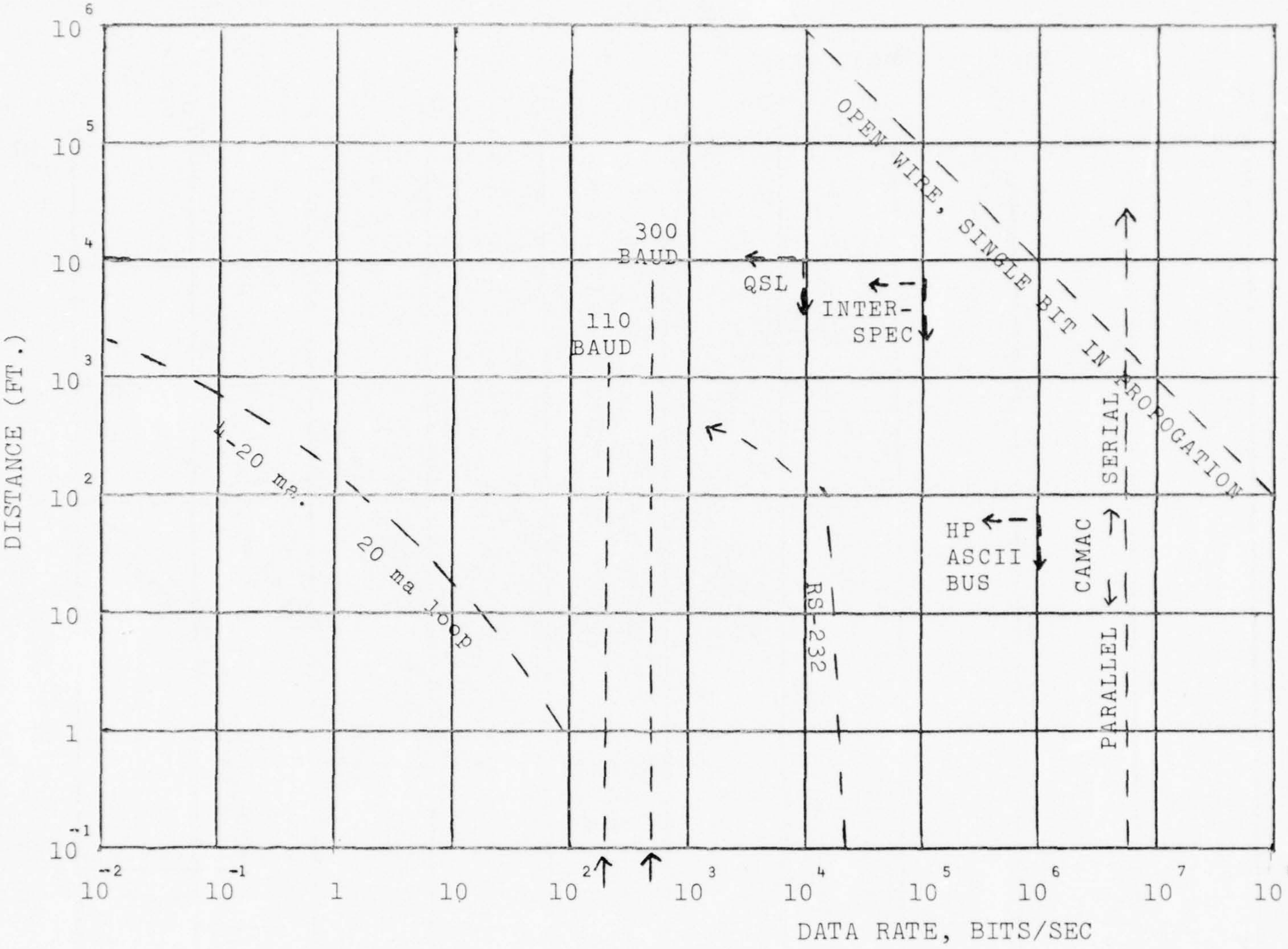


FIGURE 2

TRANSMISSION CAPABILITIES OF VARIOUS APPLICABLE PROCESS-CONTROL DATA TECHNIQUES

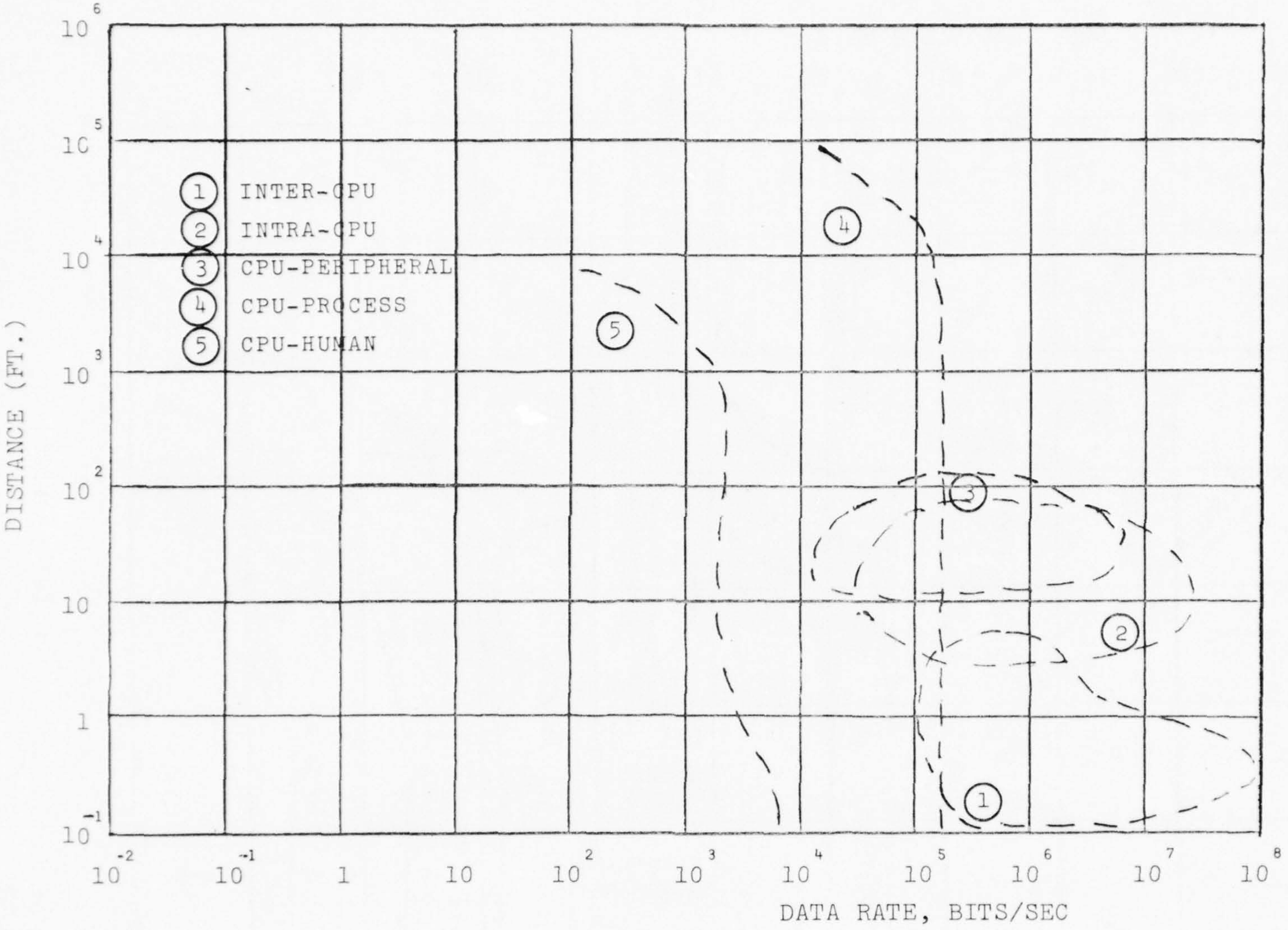


FIGURE 3

NEEDS FOR DATA TRANSMISSION CAPABILITIES FOR COMMUNICATION BETWEEN SYSTEM ELEMENTS OF A CONTROL SYSTEM

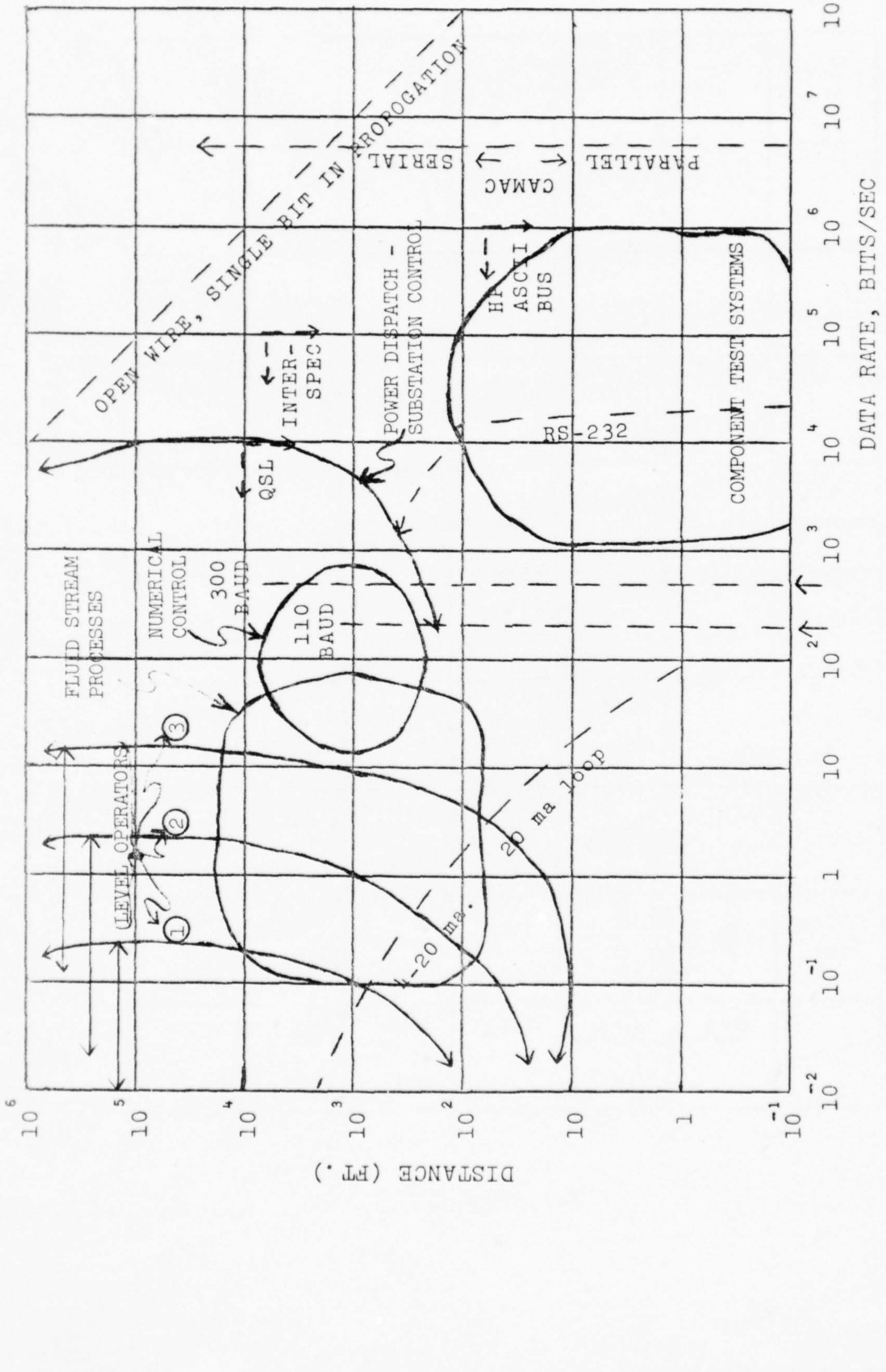


FIGURE 4

APPLICABILITY OF SOLUTIONS TO PROBLEMS
(OVERLAY OF FIGURE 2 ON FIGURE 1)

INTERNATIONAL PURDUE WORKSHOP ON INDUSTRIAL COMPUTER SYSTEMS

PURDUE LABORATORY FOR
APPLIED INDUSTRIAL CONTROL
102 Michael Golden
Purdue University
West Lafayette, Indiana 47907, USA
317/494-8425

December 25, 1975

Please reply to:

DISCUSSION OF FUNCTIONAL REQUIREMENTS OF INTERFACE AND DATA TRANSMISSION

T. Tohyama

1. Introduction

At the recent meeting of Process Interface Committee in Tokyo the followings were discussed.

(a) What is industrial computer system and required characteristics

(b) What is the needs and characteristics of a line sharing communication

(c) Functional requirements for industrial process control inter-subsystem communication

2. Scope of work

The goal of present work is to establish the functional requirements of a general-purpose communication subsystem for Information interchange between subsystem of a computer-based process measurement and control system.

Any specific standard of a general-purpose communication subsystem shall be evaluated according this functional requirement.

Thereafter, a standard of industrial process control computer inter-subsystem communication must be well defined.

3. Application environment

The communication subsystem is to be used primarily in the industrial process control computer system.

INDUSTRIAL PROCESS CONTROL COMPUTER

* A computer should be capable to be utilized for closed loop process control

* A industrial process should be kept up without relation of computer start and stop (A industrial process can not operate with synchronization of computer running)

* A industrial process is to produce material or energy change

Note; Typical application areas of industrial process control computer are :

- Petroleum and chemical process
- Iron and steel process
- Power generation process
- Utility industries
- Integrated machine tool plants (DNC)

The following applications are not included in generally:

- Labotatory automation
- Traffic automation
- Building automation
- Mechanical automation

4. Subsystem types

On-line real time communication are requires between subsystems of the following types:

(a) Process input and output interface

(b) Man-machine communication interface

(c) Computer communication interface

(d) Service and support equipment interface

5. Basic requirement

REQUIREMENTS

(1) The proposed communication shall be a information interchange between distrbuted subsystems of a industrial process control computer

(2) The communication shall be a serial system

(3) The communication shall be a line sharing system

(4) The communication shall be independent of any characteristics unique to a particular subsystem or devices

(5) The communication and interface shall be achieved high reliable operation for industrial process environmental conditions. For high reliability, the followings shall be covered :

(5.1) High reliability of hardware

(5.2) High reliability of information interchange
(very small error rate in communication line)

(5.3) The system shall be operable within a industrial process environmental conditions involving noisy condition and so on

(6) The communication and interface shall provide for safety and security capability For failure protection, the followings shall be covered :

(6.1) Error detection capability

(6.2) Error recovery and protection capability

(6.3) Failing subsystem (or station) should not impair there subsystem (or station) or prevent them line sharing operations

(7) The communication and interface shall be reflected an appropriate trade-off between communication efficiency and system cost

(8) The communication subsystem shall be capable in the high speed and efficiency For efficient communication, the followings shall be included :

(8.1) Transmission format efficiency

(8.2) High response

(8.3) High throughput

(9) The communication subsystem shall be ease of use as following points :

(9.1) Simple architecture and mechanism

(9.2) Ease maintenance capability

(9.3) Good testing and fault diagnosing capability

(9.4) Good docummentation

(10) The communication subsystem shall be flexibility to be sufficient to support some redundancy in system design, expansion and modification

(11) The communication subsystem shall support data transmission more than 2 Km distance

(12) The communication shall be capable to handle demand (asyncronous input or interrupt input).

(13) The communication subsystem shall be code transparency in the data field.

(14) The communication shall support the following subsystems :

(a) Process I/O interface

(b) Man-machine communication interface

(c) Computer communication interface

(d) Service and support equipment interface

6. Proposal requirements

According basic requirements ; the following requirements of communication subsystem are necessary for the future discussion.

(1) Data transparency ; Support the ability to transmit uniformatted binary data and byte oriented data

(2) Priority interrupt handling ; Support the ability to get or give an asynchronous data with priority within the time limits

(3) Message-reply transmission sequences by using self controlling message ; The transmission procedure shall be due to message-reply transmission sequences The message itself shall include information text, related control information and/or control commands

(4) Detect garbled data ; Support the ability to detect garbled data as such at the receiving and so that the receiving subsystem can ignore it and error recovery procedures can be initiated

(5) Error recovery ; Support the ability to prepare error recovery procedures which, to the greatest extent possible, are automatic

(6) Data block ; Handle efficiently data block of widely different lengths

(7) Avoidance of unnecessary bit overhead ; Support the high transfer efficiency

(8) Dissappearance of subsystem in-mid-message ; Cope with the absence of an addressed subsystem, and failure in any subsystem does not impair other subsystem or prevent them from line sharing operations

(9) Logically complete ; Every possible transaction sequence must be predictable in its outcome and it must exit to an acceptable state. Logical completeness may be demonstrated by a complete transition state analysis

(10) Buffering ; The transmission procedures shall be operated in a fully buffered autonomous mode

(11) Subsystem remove ; Support the ability to put a subsystem online or removing it does not disturb the correct function and operation of other subsystem

(12) It is not necessary to be closed loop communication line (It is better to be branch way communication line)

(13) It is not necessary to be fixed control station

A COMPARATIVE LOOK AT INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSAL TO IEC
JEIDA DATA HIGHWAY
PURDUE EUROPE
CAMAC SERIAL
ISO HDLC

G. MERCKEL
GENERAL SYSTEMS DIVISION
IBM CORPORATION
BOCA RATON, FLORIDA

INDUSTRIAL PROCESS COMPUTER INTERFACES

SUMMARY

PRESENTLY THERE ARE A NUMBER OF STANDARDS GROUPS CONSIDERING INDUSTRIAL COMPUTER SYSTEM COMMUNICATIONS. ONE OF THE LATEST[1] IS A FUNCTIONAL REQUIREMENTS STATEMENT SUBMITTED TO THE INTERNATIONAL ELECTROTECHNICAL COMMISSION (SC 65A WG6). OTHER PROPOSALS TO DATE INCLUDE THOSE BY:

- JAPAN ELECTRONIC INDUSTRY DEVELOPMENT ASSOCIATION (JEIDA) PROCESS INTERFACE COMMITTEE.
- INTERNATIONAL PURDUE WORKSHOP ON INDUSTRIAL COMPUTER SYSTEMS, EUROPE, TC5 INTERFACES AND DATA TRANSMISSION.
- EUROPEAN ESONE DATAWAY WORKING GROUP, CAMAC SERIAL
- INTERNATIONAL STANDARDS ORGANIZATION (TC 97/SC6), HIGH LEVEL DATA LINK CONTROL (HDLC)

THESE PROPOSALS ARE IN DIFFERENT STAGES OF DEVELOPMENT AND, THUS, VARY IN THEIR EXTENT OF DETAIL AND CONTENT. FOR EXAMPLE, THE CAMAC SERIAL SPECIFICATION ENCOMPASSES NOT ONLY ELECTRICAL AND MECHANICAL RECOMMENDATIONS BUT ALSO THE LINE PROTOCOL. HDLC, ON THE OTHER HAND SPECIFIES ONLY THE LINE PROTOCOL REQUIRED FOR INFORMATION TRANSFER AND LINK CONTROL. THE JEIDA AND PURDUE EUROPE PROPOSALS ARE SUMMARY IN NATURE, BOTH BEING RELATIVELY NEW WORK.

NEVERTHELESS, EMPLOYING THE PROPOSAL TO THE IEC AS A BASE, A COMPARATIVE ANALYSIS OF THE OTHER CURRENT PROPOSED STANDARDS WAS CONDUCTED AND IS ATTACHED. A FEW ADDITIONAL COMMENTS ON SYNCHRONOUS DATA LINK CONTROL HAVE BEEN INCLUDED WITHIN THE HDLC NARRATIVE.

INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSED IEC REQ.'S	JEIDA DATAWAY	PURDUE EUROPE	CAMAC SERIAL	ISO HDLC
1.0 INTRODUCTION				
. General purpose communications system for information interchange between subsystems of a computer based process measurement and control system.	. Requirements for an industrial dataway including scope, usage, comm. cable, protocol, transmission method and synchronization, coupling method between station and comm. line.	. A communications system for process control applications.	. A logic structure for message transfer, independent of the speed of transmission for low/intermediate response time applications (definition includes electrical/mechanical/protocol).	. A line discipline or protocol for the management of information transfer over a data communications channel.
2.0 APPLICATION ENVIRONMENT				
. Used in process industry characterized by on-line, real time system. . Requires secure and usually dedicated channels implying an intra-plant cable system.	. Used in industrial system characterized by real time.	. Real time operation with staggerec efficiency and function (not intended for telecommunications, high speed processor to processor comm., decentralized mapping of computer I/O channels for peripherals).	. Time-sharing real time system for in-plant monitoring and control.	. A data link control discipline for serial-by-bit transmission between buffered stations on a data transmission link using centralized control.
3.0 SUBSYSTEM TYPES				
. Communication between pairs of operator consoles, process I/O, terminal storage systems, data entry sys., computers, data output PRT, CRT, Alarm and be able to accommodate peripherals - TTY, PRT, ...	. Connect multi-data sender and receiver in one data comm. line. . Connect computer and many types of process interface and/or peripheral devices.	. Connect stations which can reply and generate asynchronous demands, stations which can only reply.	. Connect stations which can reply and generate asynchronous demands.	. Connect stations which can reply and generate demands on a polling basis. Station transmission w/o permission allowed.

INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSED IEC REQ.'S	JEIDA DATAWAY	PURDUE EUROPE	CAMAC SERIAL	ISC HDLC
4.0 CAPABILITIES				
. Support centralized, hierarchical distributed, hybrid configurations . Capable of providing redunant electronics for sys. availability . Ideally support self-repairing capability . Direct station-to-station communication . Buffered autonomous mode-once data accepted by comm. sys., data transferred w/o error . Capable of change or expansion after installation . Transparent to distance	. Loop configuration . Communicate with each device 1:m or n:m mode (between any local stations) . Adding/deleting a station possible w/o system operational disturbances	. Loop configuration . Hierarchically organized control system with high reliability . Fixed master station at one time . Adding/deleting a station on line should not disturb correct operation of other stations	. Loop configuration . Communications channel undefined (defined ports)	. Channel configuration - point to point, multipoint . Transmission modes - HDX FDX on switched, non-switched networks (SDLC provides also for loop and hub configurations) . Primary to secondary communication, no secondary to secondary comm. directly . Primary station control . Communications channel undefined . Transparent to variation in line speeds/propagation delays
5.0 COMPATIBILITY WITH OTHER SYSTEMS				
. Capable of interfacing with common carrier channels for remote process I/O-interface with ele/mech. compatibility to RS XYZ.				

INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSED IEC REQ.'S	JEIDA DATAWAY	PURDUE EUROPE	CAMAC SERIAL	ISO HDLC
6.0 MAINTENANCE AND SERVICEABILITY FEATURES				
·Provide test and fault diagnosis - includes traffic monitoring and both local and remote loopback test facilities ·Allow computer IPL over data channel		·Provide fcr duplex master station option	·CMDS provided for control/status checking of stations	·Frames for station control/status
8.0 SAFETY AND OTHER STANDARDS				
·Meet all pertinent mandatory standards of licensing agencies				
9.0 TRAFFIC CONSIDERATIONS				
·Traffic characteristics should be formulated and stated	·Effective transmission speed ≥ 100K bit/sec	·Information flow at ~130K bits/sec	·Clock at ≤ 5MHZ	·Unspecified
10.0 AVAILABILITY, RELIABILITY, DATA INTEGRITY				
·No loss of sys funct with one-point line cut ·Station failure must not cause comm. sys failure ·Significant failure modes must not be catastrophic ·Error detection in both direction for status/control ·Maintain data integrity/sequencing in noisey environment	·Station failure should not affect function of comm. sys	·Failing station should not influence comm. sys integrity	·Station may be placed into off-line state w/o affecting comm. sys ·Station by-pass and loop collapse options ·Error detection in both directions for status/control through transverse and longitudinal parity	·Error detection in both directions for status/control through CRC per frame

INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSED IEC REQ.'S	JEIDA DATAWAY	PURDUE EUROPE	CAMAC SERIAL	ISO HDLC
7.0 PROTOCOL				
• Call Establishment and Release • Transmitter/Receiver(s) Identification • Data Transfer with Accuracy Verification • Every Transaction Must De Predictable In Its Outcome	• Self Controlling MSG	• Command/Reply Sequence • MSG ACK/NAK	• Command/Reply Sequence • MSG Self Controlling	• Acknowledgement Required • Frame Self Controlling • Transmission Block Length Independent of Record Length • Error Recovery-Retransmission, Status Reporting
• Error Recovery - Detect Bad Data At Receiver and Initiate Recovery Procedure - Error Recovery Procedures To Be Automatic As Far As Possible - Handle Station Failure In MID-MSG (Pwr Failure), Absence of Addressed Station	• Capable Of Handling Error Detection and Recovery	• Transmission Error Protection, Detection (Check Bytes)	• Byte Odd Parity, MSG Longitudinal Even Parity • Echo Check (Header back to Primary) • Timeouts/Retransmission • NRZL (Break Up O's)	• Recovery Discipline Capable of being Automatic • Error Prevention - CRC for Adr/Cntrl/Info - Send/Receive Count - Zero Dit Insertion (SDLC - NRZI, Timeouts) • Code Transparency • Information Field Variable • Open Ended Adr/Cntrl Structure
• Data Tranparency • Variable MSG Lengths • Open-Ended Addressing/ Control Structures • Response Time Guarantee Depending on MSG	• Code Transparency in Data Field • Handle Block Data Transfer • Byte Oriented MSG • Respond to Demand-Request ≤50 MS • Handle Priority Interrupts	• Code Transparency In Data Field • Block Transfer Possible • Byte Oriented MSG • High Transfer Efficiency for Short Messages (Avoid Long Blocks in General) • Demand Handling Possible • Reaction Time ~10 MS	• Block Transfer Not Possible (under consideration) • Byte Oriented MSG • Standard MSG Lengths to insure high response rates • Addressing/Control Structures Fixed • Asynchronous Demands Possible	• Asynchronous Response Mode allows Station to Transmit w/o Primary Permission (SDLC - Loop config. demands solicited on Poll Cycle)

INDUSTRIAL PROCESS COMPUTER INTERFACES

PROPOSED IEC REQ.'S	JEIDA DATAWAY	PURDUE EUROPE	CAMAC SERIAL	ISO HDLC
OTHER				
	· Bit Serial Transmission	· Bit Serial Transmission · Closed Loop with one line, two line option · Transmission procedures should not require delay buffers in line · Bit synch by self clocking techniques preferably from master clock · Byte delimiter bits should not be provided · Distance ~1.5 KM to 5 KM max · Word length 16 bits ~~≏10~~ bits · No. I/O points ≏ 2000 · MTBF ≏ 7000-8000 hrs · Trans. element = 8 bit byte · Separation of functional and physical specifications . Galvanic isolation at each station	· Bit Serial or Bit Parallel Byte Serial Transmission · Closed loop, two line option . Delay buffer of 3 bytes . Bit synch by primary clock, byte synch by delimiter bits, MSG synch by wait bytes. · Defined MSG structure includes device sub/adr · MSG types - - Command (5 or 9 bytes) - Reply (3 or 7 bytes) - Demand (3 bytes)	. Bit Serial Transmission . Min frame is 48 bits plus 0 bit fills . Frame types - - Information (Data) - Supervisory (Flow control) - Link control . No provision for - internal record delimiters - device support at secondary stations - Procedures to establish, maintain, terminate switched channel connections - Supervisory signal exchange between modems

REFERENCES

1. "FUNCTIONAL REQUIREMENTS FOR INDUSTRIAL PROCESS COMPUTER INTER-SUBSYSTEM COMMUNICATION", LETTER FROM J. LEE, FOXBORO TO J. A. HRUSKOCI, INLAND STEEL, SEPTEMBER 23, 1975.

2. "PRESENT WORKING FOR DATA HIGHWAY", LETTER FROM T. TOHYAMA, CHAIRMAN PROCESS INTERFACE COMMITTEE, JEIDA TO T. WILLMOTT, FOXBORO, DATED MAY 26, 1975. JAPAN ELECTRONIC INDUSTRY DEVELOPMENT ASSOCIATION.

3. "A BIT SERIAL LINE SHARING SYSTEM FOR PROCESS CONTROL UNDER REAL TIME CONDITIONS", WORKING PAPER PURDUE EUROPE, TC5 INTERFACES AND DATA TRANSMISSION, MARCH 1, 1975.

4. "SERIAL LINE SHARING SYSTEM FOR PROCESS CONTROL UNDER REAL TIME CONDITIONS", WORKING PAPER PURDUE EUROPE, TC5 INTERFACES AND DATA TRANSMISSION, JULY 11, 1975.

5. "CAMAC SERIAL SYSTEM ORGANIZATION - A DESCRIPTION", UNITED STATES ATOMIC ENERGY COMMISSION, TID-26488, DECEMBER, 1973.

6. "DATA COMMUNICATION - HIGH-LEVEL DATA LINK CONTROL PROCEDURES - FRAME STRUCTURE", DRAFT INTERNATIONAL STANDARD ISO/DIS 3309-2, JUNE, 1975.

7. "DATA COMMUNICATION - HIGH-LEVEL DATA LINK CONTROL PROCEDURES - ELEMENTS OF PROCEDURES", DRAFT INTERNATIONAL STANDARD, DOC. 1005, MAY, 1975.

REFERENCES (CON'T.)

8. "IBM SYNCHRONOUS DATA LINK CONTROL GENERAL INFORMATION", IBM MANUAL GA27-3093-0, FIRST EDITION, MARCH, 1974.

9. "SYNCHRONOUS DATA LINK CONTROL: A PERSPECTIVE", R. DONNAN AND J. KERSEY, IBM SYSTEM JOURNAL, NO. 2, 1974.

10. "IBM PROTOCOLS PART 2: SDLC", J. BUCKLEY, COMPUTER DESIGN, FEBRUARY 1975.

SECTION II

GUIDELINES AND RELATED DOCUMENTS OF THE MAN/MACHINE COMMUNICATIONS COMMITTEE

The major activity of the Man/Machine Communications Committee of the Workshop to date has been the production of its *Guidelines for the Design of Man/Machine Interfaces for Process Control* which was published as a separately bound document by the International Purdue Workshop on Industrial Computer Systems in June 1976. This document is included separately in this set of summaries. Also included are several of the background documents developed by members of the Committee and used in the preparation of the *Guidelines*.

These latter are as follows:

1. "Man-Machine Communication Guidelines", *Minutes First Purdue Workshop on Standardization of Industrial Computer Languages*, Insert IX, pp. 67-73.

2. "Specification, CRT Trend Recording System", *Minutes Second Purdue Meeting*, *ISA Computer Control Workshop*, Insert V-1, pp. 41-61, by Ronald L. Gornick.

3. "Standard Operator's Console Guidebook - JEIDA 17", *Ibid*, Insert IV-2, pp. 21-37.

4. "Future Operator Consoles for Improved Decision-Making and Safety", *Ibid*, Appendix III, pp. 131-136, by R. Dallemonti, reprinted from *Instrumentation Technology*, August 1972.

MAN-MACHINE COMMUNICATION GUIDELINES

The guidelines on Attachment A were used in conducting the discussion of communication requirements for the everyday use of an industrial computer system by the First Workshop. Attachment B presents a set of console functions developed in the same discussion.

Attachment A

COMMITTEE REPORT

MAN-MACHINE COMMUNICATION

A. THE PROBLEM

OPERATION AND MAINTENANCE OF AN INDUSTRIAL COMPUTER SYSTEM

- Continuous
- Batch
- Laboratory
- Management Information
- Background/Foreground Environment

B. THE USER

PROCESS OPERATOR

TECHNICAL AND NON-TECHNICAL SUPERVISION

CONTROL ENGINEER AND MAINTENANCE

SYSTEM MAINTENANCE

BACKGROUND APPLICATIONS

C. STANDARDIZATION GOALS

COMMUNICATIONS FUNCTIONS

COMMUNICATION DEVICES

VENDOR SOFTWARE SUPPORT OF FUNCTIONS AND DEVICES

D. FUTURE PLANS

ADDITIONAL DEVICES

- Audio Response
- Optical Character Reader
- Graphic Display
- Microfilm Projection
- X-Y Plotter

FURTHER CONSIDERATION OF ALPHANUMERIC IDENTIFICATION OF

Variables

Control Loops

Functions

ALARM DISPLAY ORGANIZATION

Consideration of improved means of generating alarms for operator guidance and/or computer action.

Consideration of basis for organizing display so that alarms are meaningful aids even under transient conditions when too many alarms occur for individual consideration.

The committee has developed the following desirable general capabilities of the programming system to describe the general man-machine communication requirements, and Tables I and II which describe the desired functional and device requirements. The committee recommends the adoption of these rules, and functional and device requirements as a standard for man-machine communication with Industrial computer systems.

1. Parameters to be entered and displayed through the communication system should be specified by function: scan, alarm, control, log, etc.
2. All entires should be displayed before entry.
3. All entries which change parameters, or alter operations should be recorded. The record is to be from the stored location, not from the entry device.
4. All alpha numeric demand displays will have the capability of also being recorded: DISPLAY; DISPLAY/WRITE. A CRT display should also follow this rule.
5. The communication system, will provide the capability of displaying, trending, and entering new engineering and calculated values.

5.1 Display: instantaneous vs. continuous update

5.2 Trend

5.2.1 Trend recorder (multi-pen)

5.2.2 Trend typewriter (alterable list of variables)

5.2.3 CRT

5.3 Enter New Value when point is removed from scan and to allow or disallow the processing of a manually entered value.

While Tables I and II describe Functions and Devices, recognition must be given to the implied software and language support required to use these Functions and Devices. Attachment B is an example of a set of typical console Functions for an industrial computer. No attempt should be made to interpret this as complete or as a standard configuration. It is also recognized that the console required is a function of the size and complexity of the particular industrial system.

Further work on man-machine communications should include study of the need for additional functions and devices. However, the work should be coordinated with the groups (especially Committees 2 & 3) and should take place after those groups are further developed.

TABLE I

MAN-MACHINE COMMUNICATION FUNCTION

FUNCTION	TYPEWRITER					LINE PRINTER	TREND PEN	OPER PEN	CRT W/KB	LIGHTS		PAPER TAPE		CARD	
	I/O	OUTPUT													
	PROG	ALARM	LOG	MSG	TREND LOG					STATE	SEQ	IN	OUT	IN	OUT
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PROCESS OPERATOR															
Data Acq. Parameters	E W	W		W		W		E D	E D			E		E	
Control Parameters	E W	W		W		W		E D	E D	D	D	E	E	E	
Computer-Manual Changeover	AIR			W		W		A I R	A I R						
Input Scan: A/L, Digital	W A I S			W		W		A I S	D A I S						
Alarm Scan: A/L, Digital	W A IS	W		W		W		A I S	D A I S	D					
Analyser: Calib. & A,I	W A I S			W		W		A I S	D A I S						
Demand Log	A W		W			W		A	A D						
Trend Log					W			A I	A I D						
Trend Record: Multi-pen	W A I			W	W		W	A I	D A I						
Trend Digital Display	W A I							D A I	D A I						
Control Loop	AISW			W		W		A IS	AISD						
Calender; Time of Day	W A IS			W				D AIS	D A I S						
System Test Programs Peripherial Devices Instrument Test Computer Start-up	W A IS			W				A I S	D A I S			AI		AI	
Interactive Syst. Data	E W A R			W				E D R	E D R						
Alarm Action	AIRW	W		W				A I R	AIRD						
Appl. Programs	AIRW			W				A I RS	A IRDS			AI		AI	

E - Enter	A - Activate
D - Display	I - Inactivate
W - Write:Hard Copy	R - Respond
P	S - Status

MAN-MACHINE COMMUNICATION FUNCTION

FUNCTION	TYPEWRITER					LINE PRINTER	TREND PEN	OPER PEN	CRT W/KB	LIGHTS		PAPER TAPE		CARD	
	I/O	OUTPUT													
	PROG	ALARM	LOG	MSG	TREND LOG					STATE	SEQ	IN	OUT	IN	OUT
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
NON-TECHNICAL AND TECHNICAL SUPERVISION															
Update	EAW			W		W		ED A	E D A			E A		E A	
Error Recovery	E W A R			W		W		E A R	E A R			E A		E A	
Calculation Program	E W A I			W				E A I	E A I			E A		E A	
Mgt.Info.System	E W A I			W		W		E A I	E D A I			E A	P	E A	P
CONTROL ENGR. & MAINT.															
Control Loop Tuning	E W A			W				E D A S	E D A S			E A		E A	
Inst. Test: A/L & Digital	E W A			W				E D A S	E D A S			A		A	
BACKGROUND APPLICATIONS															
On-Line Debug	E W A			W				E D A	E D A			E A		E A	
On-Line Compiler	E W A			W		W			E D A			E A	P	E A	P
Batch Prog. Proc.	E A					W						E A		E A	
SYSTEM MAINTENANCE															
System Programs Update	E A W					W			E D A			E A		E A	
On-Line Modification	E W A					W			E D A			E A		E A	
Core, Bulk Dump	E A W					W			E D A				P		P
On-Line Diagnostics	E W AI					W		E D A I	D A I			E A		E A	
Off-Line Diagnostics	E W A I					W			D AI			E A		E A	

E - Enter A - Activate I - Inactivate W - Write:Hard Copy R - Respond S - Status P - Punch

TABLE II

ATTACHMENT B

EXAMPLE OF
CONSOLE FUNCTIONS

A. ALPHANUMERIC DISPLAY CAPABILITY

Sufficient to conform to ISA Standard

B. PUSHBUTTON KEYBOARD FOR ALPHANUMERIC ENTRY AND FUNCTION KEYS

C. THUMBWHEEL OR DIALS FOR CONTINUOUS UPDATE VARIABLES

D. MEANS TO PROTECT AGAINST INVALUD ENTRY OF DATA

E. CONSOLE DISPLAY

Variable Name
Function
Measured Value
New Value

F. CONTROL LOOP TUNING DISPLAY

Set Point
Measured Variable

SPECIFICATION

CRT TREND RECORDING

SYSTEM

BY

RONALD L. GORNICK

ESSO RESEARCH AND ENGINEERING COMPANY

TECHNOLOGY DEPARTMENT

TABLE OF CONTENTS

CRT TREND RECORDING SYSTEM SPECIFICATION

1. INTRODUCTION

1.1 Scope

This specification describes a recording system to be used for storing and displaying graphical trends of process variables and describes responsibility the vendor must assume upon award of a contract. it is the vendor's responsibility to size and select the proper equipment to fulfill all requirements of this specification. A prime consideration is low cost on a per point basis. The vendor is, therefore, encouraged to propose as "options" any items which could significantly reduce the cost of his proposed system.

1.2 Definitions

Since many terms and phrases are interpreted differently by both vendors and users, a brief definition of terms is presented. Vendor should consider these definitions as Esso's intent when used in this specification.

- Data File

  One set of data, process parameters, process info, and status bits related to one measured process variable.

- Expandable

  Refers to additions that can be added to system after being in operation without extended (longer than 24 hours total) shutdown.

- Hard Copy

  Any suitable means of making a permanent copy of a graphic trend of a process variable.

- Operator's Console

  Refers to that section of the overall control house panel where the operator communicates with the plant through the CRT trend recorder system.

- Process Variable

  Refers to any and all transmitter signals coming back to the control house.

- Split Screen

  Refers to putting more than one variable on a CRT by having the CRT divided into separate unique divisions per variable.

- Suppressing Range

  Refers to the operator being able to change the limits of the Y-Axis for the time the trended variable is on the CRT. This allows the operator to "zoom in" on this variable's trend to permit better readability.

It is vendor's responsibility to provide hardware necessary to maintain an overall system specification (referred to ideal inputs applied to input terminals) as follows:

• Resolution	-	0.1% of full scale
• Accuracy	-	± 0.1% of full scale
• Repeatability	-	± 0.21% of full scale

1.3 System Concept

The following section briefly describes the philosophy of the system. A cathode ray tube (CRT) system is required that would:

- Permit random access to any 20 of 1000 process variables on demand through an operator's console.

- Centralize all plant trend recording.
- Provide additional flexibilities such as suppressing range.
- Provide hard copy on demand.

The type of system envisioned will have (1) an input system, (2) operator displays and consoles, and (3) CPU and bulk storage (drum or fixed head disc). It should be possible to add additional CRT's and to replace CRT's on a plug-in basis.

2. SYSTEM HARDWARE REQUIREMENTS

2.1 General

This section covers the hardware to be used in the system. Included are the input subsystem, CRT displays, and hard copy capabilities.

2.2 Input System

The input system should be capable of handling up to 1000 signals, the exact number depending on the particular refinery configuration. Each of the 1000 signals should be scanned twice a second. The majority of the signals will be volts developed across dropping resistors in 4-20 or 10-50 mA current loops, representing flows, pressures, levels, etc., For quotation purposes the vendor shall assume an 800 high level input system for this specification. The vendor will supply desired or required voltage ranges on the inputs. Vendor shall quote option to provide for 200 additional low level (0-40 mV) inputs, representing thermocouple readings (ISA Type J, K, and T).

Input system should be designed so as not to create any additional electrical paths between plant input transmitter current or voltage loops. Vendor shall assume that current loop is earth grounded and that neither side of the input circuit can be connected to earth ground.

2.3 CRT Displays

The CRT display should use a conventional "television" tube (either color or black and white). Storage type tubes shall not be used. Approximate size should be 17 inches across the diagonal for a split screen (3 or 4 variabler per screen, unique section of the screen dedicated to each variable). The exposure dose rate of the "soft" X-Ray emissions at any readily accessible point 5 cm from the surface of any CRT console shall not exceed 0.125 mR/Hr. under worst case operating conditions. Life expectancy of each CRT should be about 1 year under constant (24 hour per day) use. Display shall be designed for easy access, for replacement and repair. System should be sufficiently modular to permit plug-in addition of CRT's without software or hardware changes. Vendor should assume for this quotation a base system sized for simultaneous display of 20 variables and shall quote additional cost for incremental addition of simultaneous facilities for up to a total of 100 variables.

2.4 Hard Copy

It should be possible to obtain a hard copy of any system variable (displayed or not displayed). A fast speed strip chart recorder and some identification means for the variable name, range and time scale is the preferred method, but alternative techniques shall be considered.

3. FUNCTIONAL REQUIREMENTS

3.1 General

This section contains a description of the functional requirements for the CRT system. The history data base, variable names, data files, filtering, hard copy and restart requirements are covered.

3.2 History Data Base

A history will be kept for each of the 1000 process variables brought into the system. Each of the 1000 inputs will be stored on the same time base, while being scanned at a 2 sec. rate. The history will be stored for a length of 8 hours. The newest hour will be stored on a 15 second basis, the next 3 hours on a 1 minute basis, and the last 4 hours on a 2 minute basis. This gives 540 points per variable stored as history. A diagram of this storage of history is in Figure I. The value being stored will be in percent of full range of the instrument. The value should be changed to engineering units when displayed on the CRT.

3.3 Variable names And Engineering Units

Variable names for each input will be of a three-element alphanumeric nature, e.g. XYZ,

where X = Unit name, up to 24 different names up to 2 characteristics each.

Y = Variable type, up to 12 different types, one characteristic each.

Z = Variable number, in the range 001 to 999.

Examples of XYZ are shown in Appendix I.

Typical engineering units are given in Appendix II for illustration only. The vendor shall allow for 32 different types of engineering units having 12 characters each. Engineering units shall be filled in and assigned from a teletype.

3.4 Data Files And File Builder

Each variable should have associated with it a data file containing such information as engineering units, instrument range, scaling limits (for display), etc. It should be possible to assemble or change any of these parameters on-line from the teletype. In essence, this is an on-line file builder, which will be necessary to build the system.

3.5 Group Function

The file builder routine should also be capable of building display groups. A group will consist of a number of variables that are closely related in a process. This group would be given a tag number and all should be displayed on the designated CRT at the same time and with the same time scale.

3.6 Filtering

The system should contain an option to filter the data in bulk storage before being displayed on the CRT (filtering should be a simple first-order type, with time constant, adjustable through the console, on a per-variable basis). Range of equivalent time constant shall be 0 to 10 minutes with a resolution of 6 seconds. Filter constant shall be automatically displayed on the CRT with the variable. The vendor should be aware of the errors that can occur when using digital filtering.

The main errors occur from truncation in the filter equation due to poor storage resolution of the variables when using a small filter constant. An example is:

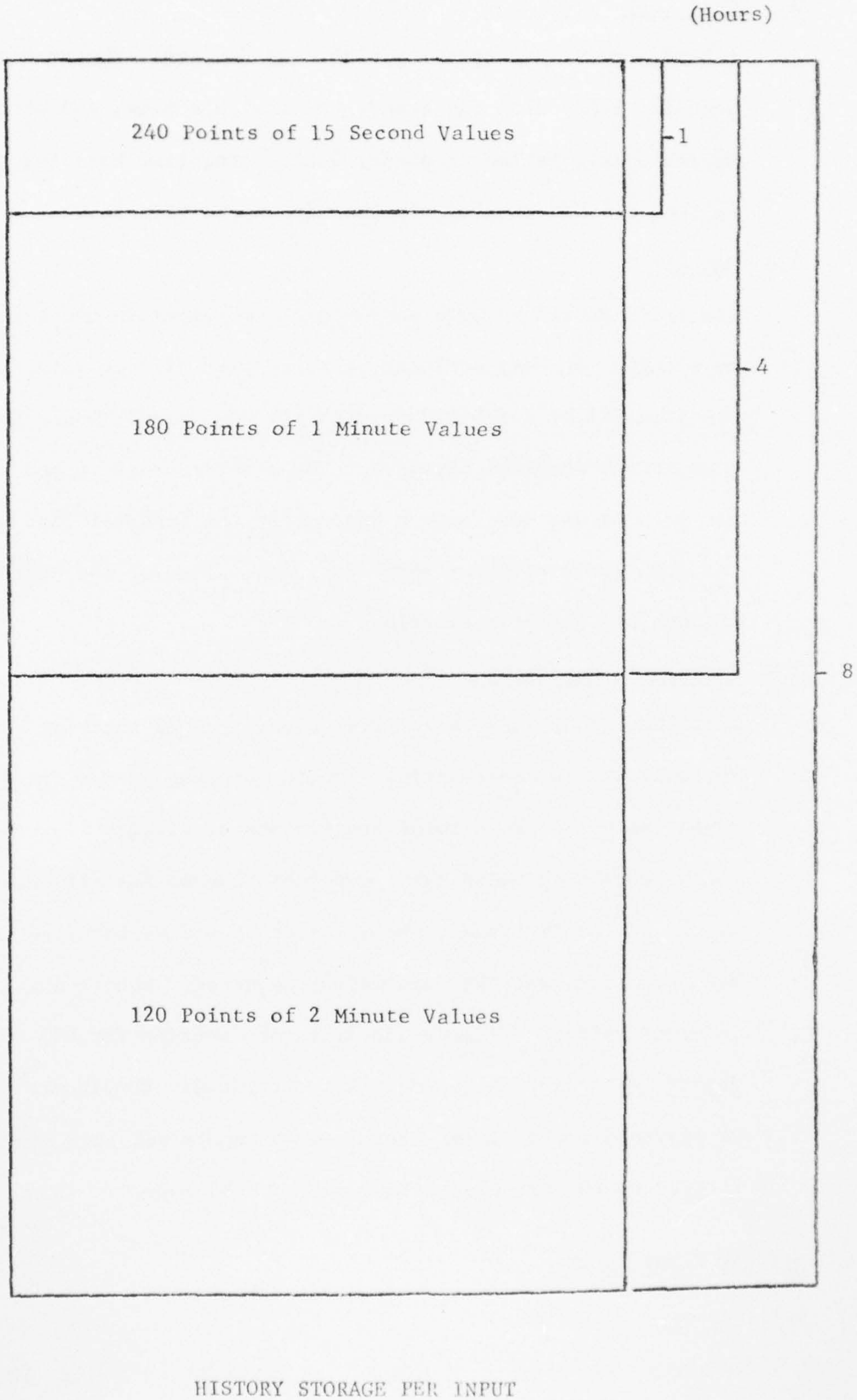
(New stored value) = (Filter constant)(Scanned value) x (1-Filter constant)(Old stored value)

Scanned value stored to .1% resolution = 90.7%

Old stored value to .1% resolution = 90.0%

(New store value) = (0.1)(90.7) + (0.9)(90.0) = 90.07 = 90.0 to .1% resolution

There was no change in the stored value, even though the new stored value would be changed with a resolution .01%.


(Hours)
240 Points of 15 Second Values
180 Points of 1 Minute Values
120 Points of 2 Minute Values
1
4
8

HISTORY STORAGE PER INPUT

FIGURE I

3.7 Hard Copy

A hard copy of any system variable is desired. The hard copy should contain, along with the trend, the variable name, and both time and variable axis labled in proper units. The time base for this should be like the CRT display, in that it should be a 24 hour- type marking.

3.8 Restart

Provision should be made to restart the system in two ways following an outage. One way would be to start with all new values, i.e. clear the drum (disk) and build up with all new values (would take up to 8 hours for complete history). The other restart method would be to use past values and leave a gap during the interval that the computer was down (primarily for short outages). Time of day would be entered through the operator.console.

3.9 Additional Functions

Additional functions that are required include checking raw values for validity (compare against instrument range), and flagging out of range values on the display (regardless of display limits), validity checks on all commands, and reasonable checks for all entered data. In the two latter cases, the operator should be notified that he has made an error, and the information rejected. Vendor shall include in quote on T.C. option a linearization routine for all thermocouple inputs (ISA type J, K and T thermocouples). Continuity checks shall be provided on low level thermocouple inputs and open circuited inputs flagged on the CRT display and on the hard copy.

4. DISPLAY REQUIREMENTS

4.1 General

It should be possible to display at least 20 variables (100 max.) at a time either on separate screens, or in a split screen fashion. The time axis should be the horizontal axis, and the veritical axis is the variable axis. An example of split-screen display is shown in Figure II.

4.2 Display Format

Each type of display will have the following common elements. The Y-axis will be labled and scaled in engineering units based on a file of scaling information for that particular variable, and the axis should be divided in at least 8 equal divisions. The time axis should also be labeled in at least 8 equal divisions. The time base shall be labled on a 24 hour basis i.e. 1 a.m. as 0100 and 1 p.m. as 1300. Each process variable display should be labeled with the variable name. The latest value on the screen should be written on the display (no interpolation required). When a screen is filled up, it should be re-displayed automatically (shifted by 10 minutes, time scale updated, and Y-axis scale same as previously displayed).

Since the scaling range will be changeable, high display and overall system resolution is desirable. When the scaling is changed, (both Y-axis and time axis) the newly designated range will continue to be displayed until the operator calls for a new change in range or the variable is taken off display. A continuous line presentation is desired; however, dots, dashes or other techniques will be considered. Any given display should take no longer than 5 seconds to appear when a new variable is assigned by the operator. Also, the minimum time between hardware scan of an input and display update should be 2 seconds.

Four types of displays are desired. The first is called a zero hour display. This display requires no history, with a 18 minute display of points updated at 2 second intervals. The total 18 minutes will be updated once every 2 seconds. When the screen is full, it should be re-displayed automatically as previously described.

The second type of display is called a one hour display. The initial 50 minutes of the display will have previous points at 15 second intervals and the subsequent 10 minute of 15 second values will be displayed in real time. When the screen is full, it should be re-displayed automatically as previously described. A sample display is shown in Figure III.

The third type of display is called a four hour display. The initial 3 hours and 50 minutes of the display will consist of previous history data taken at one minute intervals, and the subsequent 10 minutes of one minute values will be displayed in real time. When the screen is full, it should be re-displayed automatically as previously described.

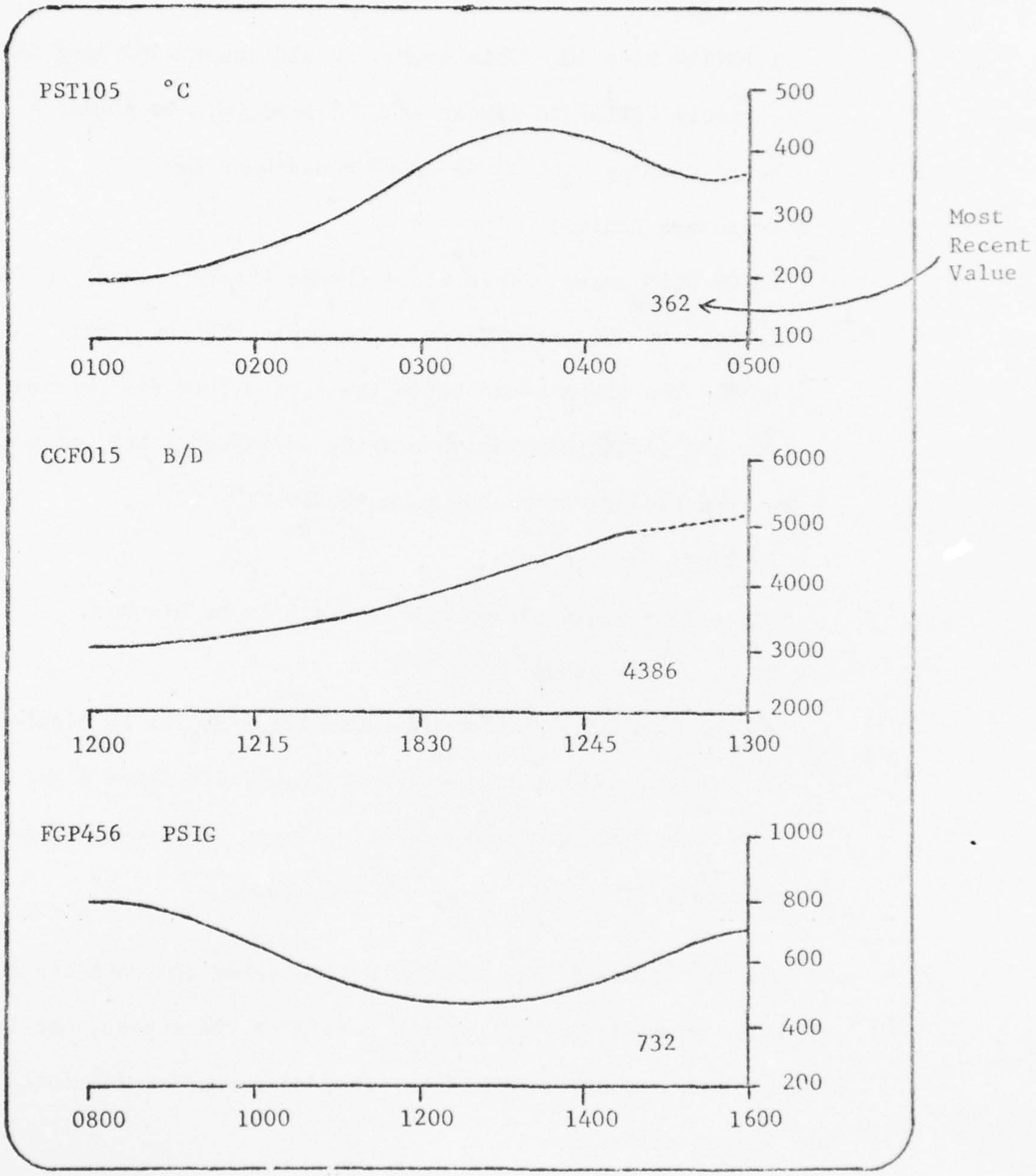
The fourth type of display is called an eight hour display. The display shall consist of 2 minute values that has no real time updating. The display shall consist of history only, and the display would stay on the CRT until removed by the operator.

(Option) The fifth type of display is called a sixteen hour display. The display shall consist of 4 minute values that has no real time updating. The display shall consist of history only, and the display would stay on the CRT until removed by the operator. This type of display would only be used for important plant variables, about 200 variables.

5. OPERATOR CONSOLE

5.1 General

CRT display assignment should be controlled via the operator console. The console shall also be used to enter information to the computer system. Two alternatives are a "standard" typewriter style keyboard, or a custom keyboard.



SPLIT-SCREEN DISPLAY

FIGURE II

5.2 Standard Keyboard

Below is an example of a format for the standard keyboard.

- To display:

  1 PST152 Disp 4C - This command would cause a one hour display of variable PST152 to appear on CRT 4 Trace C. To obtain a 'four hour' display, the first entry would be a 4.

- To change scales:

  2B CH HL1M xxxxx - This would change the high limit on CRT 2, Trace B to the value xxxxx. To change the low limit, use the word LL1M. The xxxxx would be in the appropriate engineering units for the particular variable being addressed. The scale change is only during display, not permanently.

- To clear the trace:

  Clear 1D - would cause CRT 1 Trace D to be blanked.

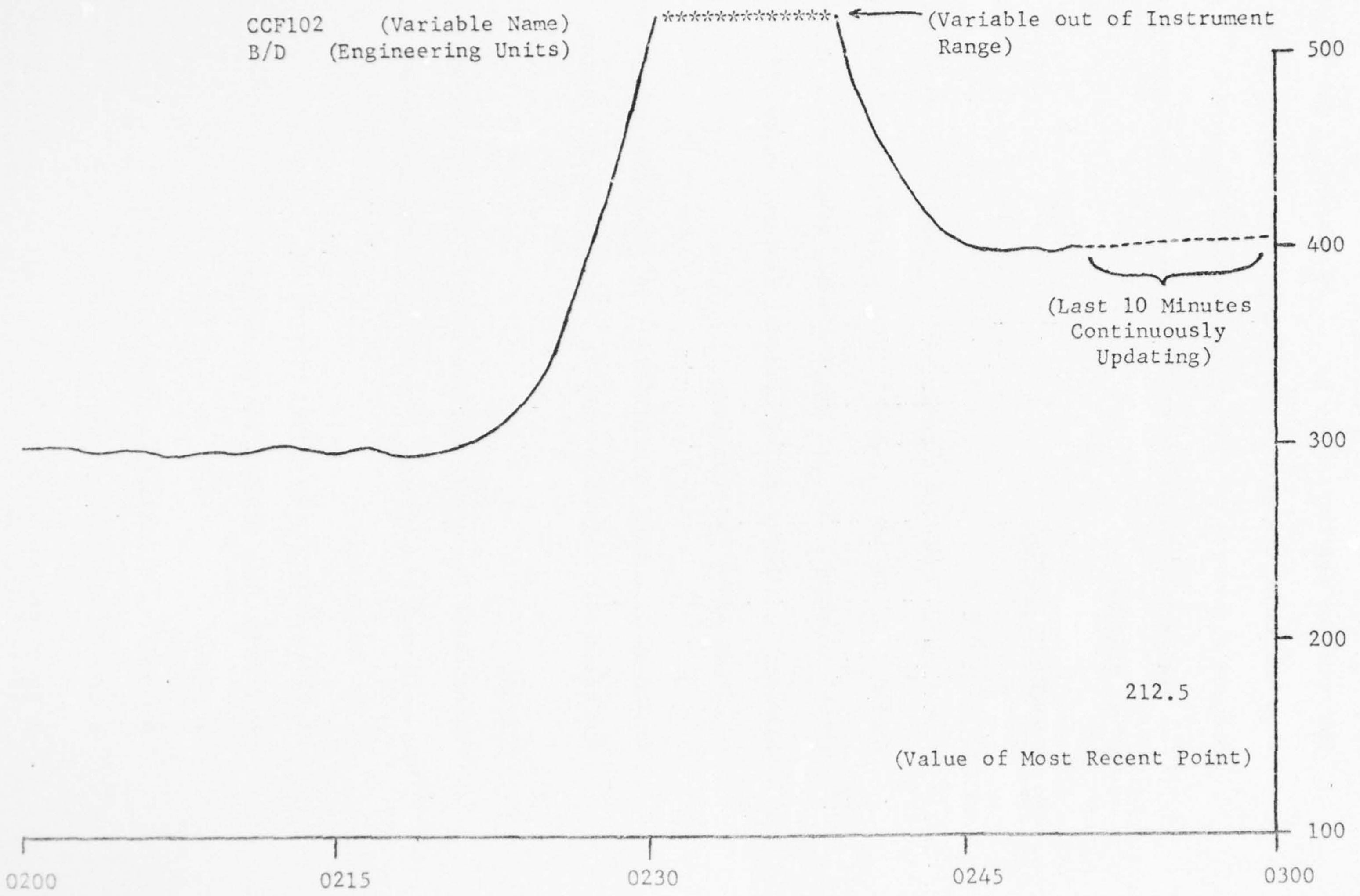
- To display a group:

  4 G563 Disp 3A - This would cause group 563 to be displayed on CRT 3, Trace A, fill in the screen starting with Trace A and going on till the whole group is displayed (even if more than one CRT is needed). They will be on a 4 hour basis.

It has been assumed that the CRT will display the commands as they are entered, perhaps at the bottom of the screen. At least 2 keyboards and CRT's should be provided for added flexibility and backup purposes.

5.3 Custom Keyboard

An alternate approach would be to provide a custom console to accomplish the above mentioned functions. An example of this type is contained in Figure IV. The trace buttons (A,B,C,D) are for a CRT that has 4 traces (if only 2 traces, A and B would only be used).



EXAMPLE-DISPLAY (1 HOUR)

FIGURE III

The nixie tube readouts serve three purposes. One is to show the operator the CRT number and trace number (9A). Another is to show the operator the tag number he called up (532). The third is to show the limit change made by the operator. Vendor is encouraged to propose alternates to this custom keyboard approach which will make the system less expensive or easier to use from a human engineering point of view.

6. MAINTENANCE AND MANUALS

Maintenance

Vendor shall indicate clearly in his proposal any items which are included in the base proposal which will facilitate maintenance and troubleshooting. If none are included, vendor shall include all necessary equipment and/or optional features which will facilitate troubleshooting and maintenance.

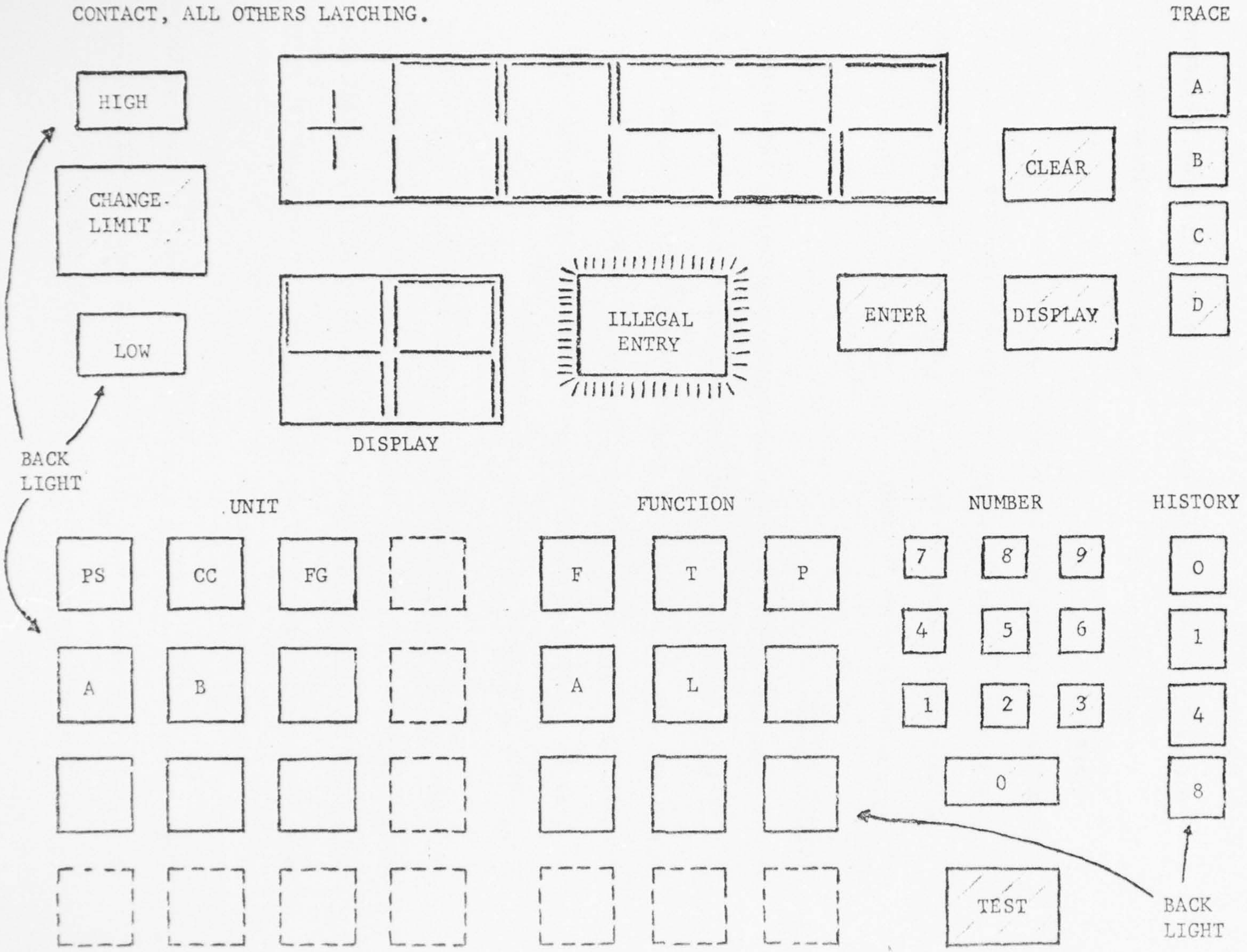
Vendor shall submit with his bid a detailed list of all test equipment and diagnostic aids necessary to troubleshoot the system.

Manuals

Vendor shall include 4 complete sets of instruction and maintenance manuals with his system. The manuals shall include but not be limited to the following:

- All necessary maintenance procedures
- Step by step calibration procedures
- Test voltage points
- Complete, up-to-date circuit schematics
- Overall simplified block diagram
- Technician level description of all schematics, block diagrams, timing diagrams, and maintenance and calibration procedures.

SHADED BUTTONS ARE MOMENTARY
CONTACT, ALL OTHERS LATCHING.



EXAMPLE CUSTOM CONSOLE

FIGURE IV

- Detailed functional description of overall system and how it operates.
- Clearly labled pictures and schematics of all cards, adjustment and calibration devices, etc. Showing physical location in system.
- List of recommended spare parts and costs
- Installation and step-by-step startup procedures

Manuals shall be considered part of system and shall be ready when system leaves vendor's plant.

7. FACTORY TESTING

Vendor shall notify Esso Research and Engineering Company Inspection Section in writing at least 10 days in advance when his system is ready for final inspection. The system is to be completely assembled, debugged, have successfully undergone vendor's quality control checkout and must have run for five (5) continuous days "hands-off" (no adjustment or failure having occurred) before final inspection and checkout.

At least 20 sets of calibrating curve data taken at intervals of 8 hours during the hands-off run shall be available for review at final inspection. Vendor shall present documentation of other final quality control check (margin voltage, common mode, ambient temperature swings, vibration tests, etc.) at time of final inspection.

Vendor shall make available a test room and all necessary test equipment for final Esso checkout. In general, the final checkout will include:

- Visual inspection for compliance with specifications
- Functional tests on entire system

- Review of vendor's quality control data and "hands-off" run data. This should include a description of quality control checks performed on the system.
- Review all drawings and instruction manuals.

Vendor shall propose test setup to be used for demonstrating computer interface.

Acceptance of system at final factory checkout does not relieve vendor of responsibility for supplying a long range reasonably trouble free, reliable system. Vendor will quote his guarantee on all equipment and his system.

APPENDIX I

VARIABLE NAMES

General Format - XYZ

Examples of X - A, B, CC, FG, PS, Pl, R2

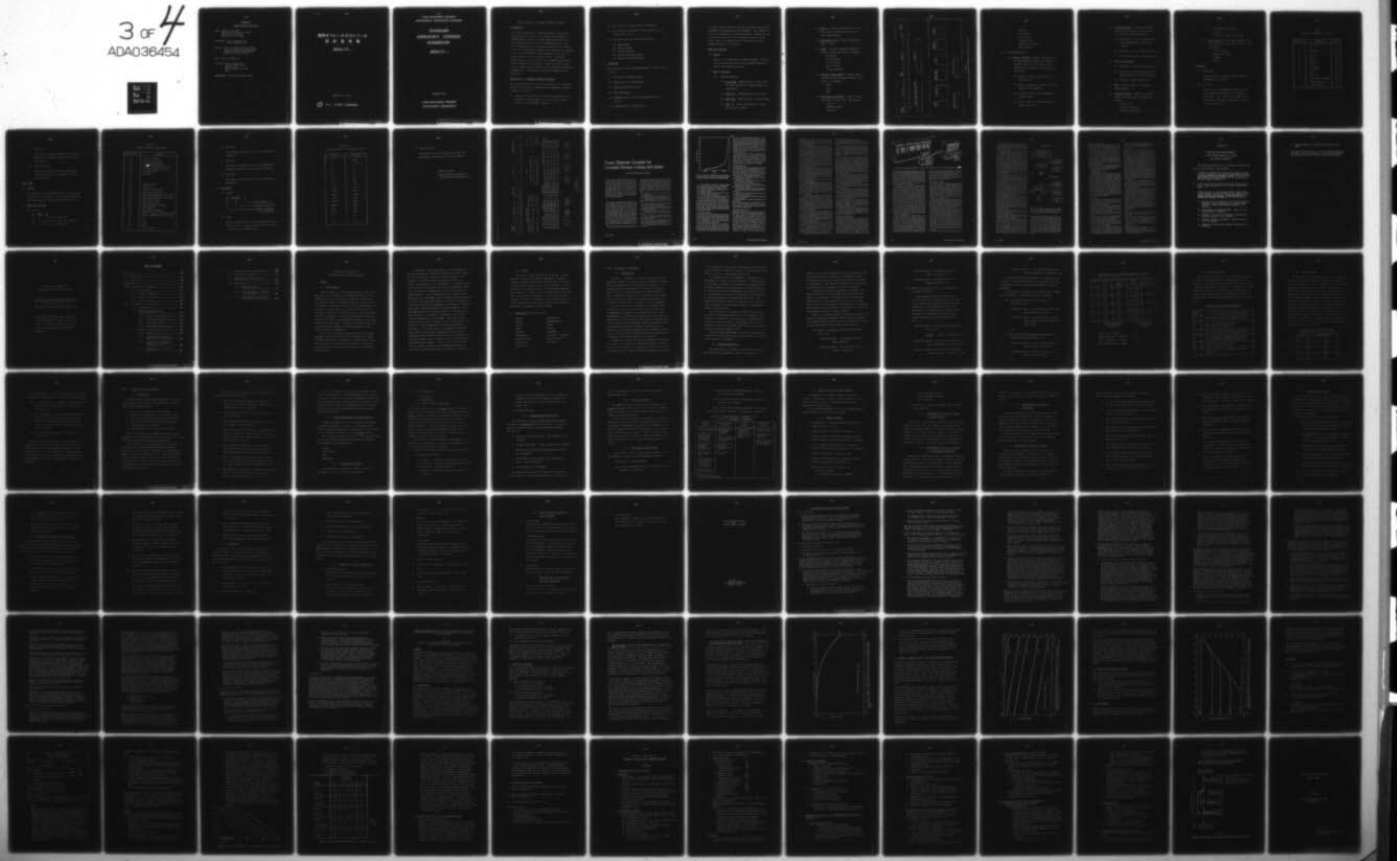
Examples of Y - F, L, T, P, A

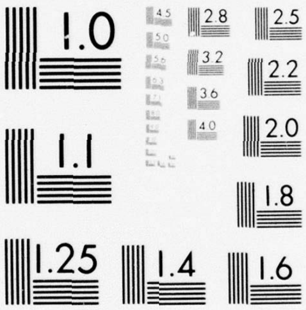
Examples of XYZ - PSF001, CCP235, FGT987

AD-A036 454   PURDUE UNIV LAFAYETTE IND PURDUE LAB FOR APPLIED IND--ETC   F/G 9/2
SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIO--ETC(U)
JAN 77   N00014-76-C-0732

UNCLASSIFIED   NL

3 OF 4
ADA036454





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

APPENDIX II

EXAMPLES OF ENGINEERING UNITS

FLOW - Barrels Per Day (B/D)
Standard Cubic Feed Per Hour (SCFH)
Gallons Per Hour (GAL/H)
Cubic Meter Per Hour (M3/H)

TEMPERATURE - Degrees Fahrenheit (°F)
Degrees Centigrade (°C)

PRESSURE - Pounds Per Square Inch Gauge (PSIG)
Pounds Per Square Inch Absolute (PSIA)
Kilograms Per Square Meter (KG/M2)
Milimeters of Mercury (MM-HG)

LEVEL - Percent of Range (%)

ANALYZERS - Percent of Range (%)
Parts Per Million (PPM)
Viscosity (CP)
Weight Percentage (Wt. PCT.)
(Ph)

MISCELLANEOUS - Revolutions Per Minute (RPM)

47 − A − 244

標準オペレータズコンソール 設計基準書

JEIDA-17-1972

昭和 47 年 7 月 制定

社団法人 日本電子工業振興協会

JAPAN ELECTRONIC INDUSTRY
DEVELOPMENT ASSOCIATION STANDARD

STANDARD OPERATOR'S CONSOLE GUIDEBOOK

JEIDA-17-1972

Established July 1972

JAPAN ELECTRONIC INDUSTRY
DEVELOPMENT ASSOCIATION

DESIGN GUIDANCE OF STANDARD OPERATOR'S CONSOLE

1. INTRODUCTION

This design guidance for standard operator's console has been prepared as a part of the computer and automation standardization service of JEIDA (Japanese Electronics Industry Development Association) established in July 1972. This standard presents the design criteria for the operator's console for communication between plant operator and computer. It will be distributed to representatives of each user or vendor company for their critical review. From their review comments, we will revise the standard after one year. In this standard, the operator's console with CRT display is not covered because we have not yet completed the detailed standardization of editing or controlling capabilities of CRT display.

2. DEFINITION OF A STANDARD OPERATOR'S CONSOLE

This standard operator's console is defined as the interface device for man-machine communication in industrial computer systems, as follows:

(1) Information communications which are necessary for the operation and management of a process plant through the use of an industrial computer.

(2) This console is manipulated by one operator.

(3) This console is installed in the operator's room (control room, etc.).

(4) This console has the following functions:

(a) Data Display

(b) Data Key Entry

(c) Function Request

(d) Status and Alarm Display

(e) Control Loop Manipulation

3. OBJECTIVES

The objectives of this standard operator's console are as follows:

(1) Efficiency of hardware design.

(2) Minimum cost of system design.

(3) Capable software functions.

(4) Easy Maintenance.

(5) Standardization of function and manipulation of console.

(6) Standardization of terminology.

For this purpose, the basic functions and their manipulation procedures are presented in this standard. This operator's console is organized in a module structure. Through use of this module structure in the operator's console, we are able to get: (1) operability; (2) reliability; and (3) reduced complexity of function.

4. NAME AND NOTATIONS

4.1 General

Figure 1 is a simplified functional diagram. Name and notations which are used in this standard operator's console are defined as follows.

4.2 Name of Function

4.2.1 Display Functions

(1) Group Name: Identification of each plant or plant location in a complex plant or a large plant.

(2) Point No.: Identification code of data.

(3) Data Type: Identification of type of data.

(4) Data (1): Value of data which is input and output to system.

(5) Data (2): Value of data which is input and output to system, mainly used for key-in data display.

(6) Engineering Units: Display engineering units of data.

(7) Alarm: Inform for abnormal condition of computer system functions, for example,

CPU Failure

Power Failure

P I/O Failure

Sensor Trouble

Illegal Operation

(8) Computer System Status: Display status of computer system and plant operation condition, for example,

Power

Busy

Scan

(9) Individual Loop Status: Display control loop status in DDC or SCC, for example,

Back Up

Computer Manual

Open Loop

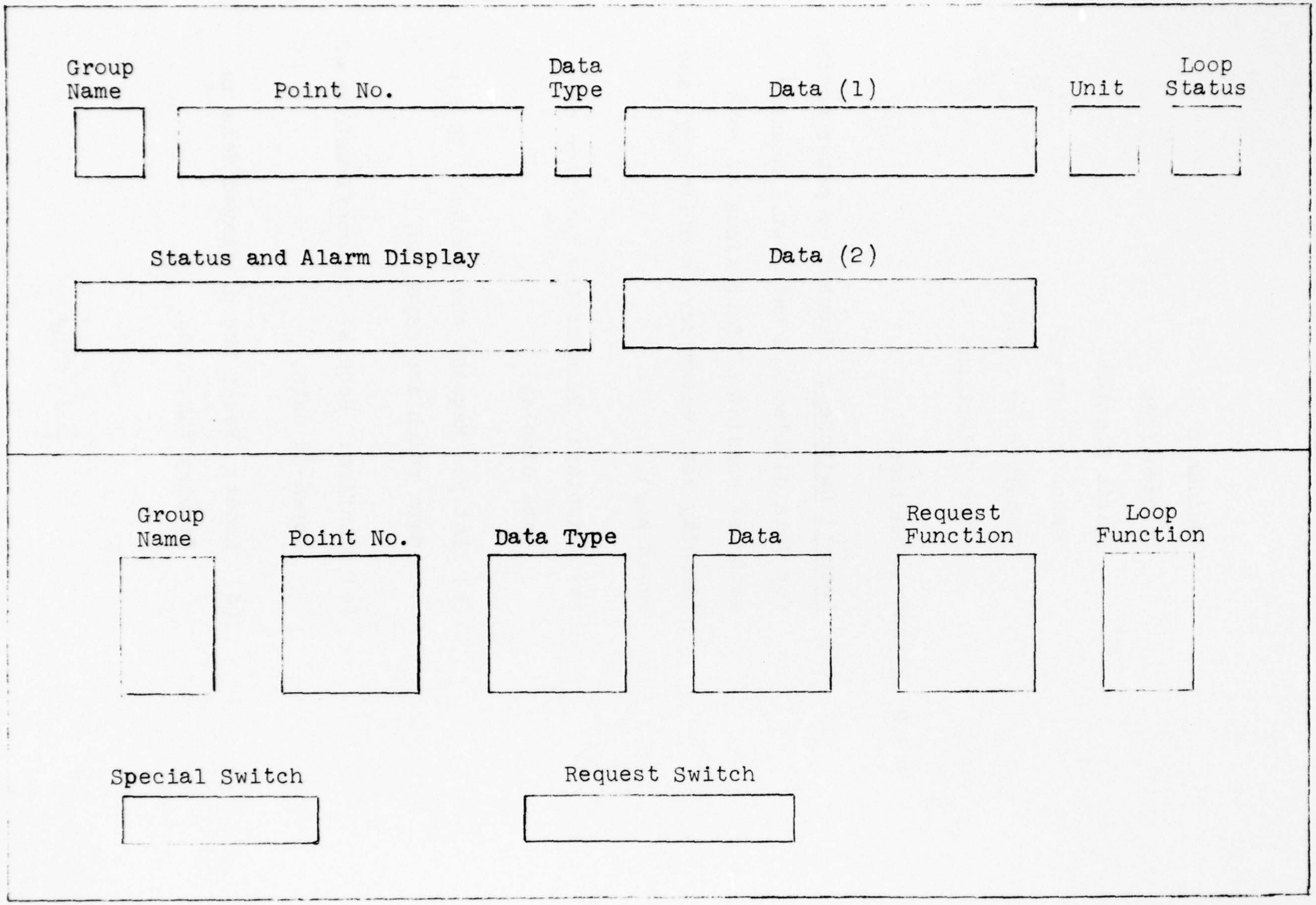


FIGURE 1

SIMPLIFIED FUNCTIONAL DIAGRAM OF STANDARD OPERATOR'S CONSOLE

Guidance

Closed Loop

Ratio Control

Cascade Control

Supervisory Control

Mode Selection

4.2.2 Keyboard Functions

(1) Request Switches: Request computer action for data display and data set. Request switches consist of four kinds of keys (display key, entry key, confirm key, and reset key).

(a) Display: Request the function of data display.

(b) Entry: Request the function to store data which are keyed in.

(c) Confirm: Request the confirmation of keyed-in data.

(d) Reset: Reset the displayed data in display functions.

(2) Special Switches: Required switches to operate the operator's console include lock, lamp test and buzzer reset.

(a) Lock Switch: Lock out certain specific manipulation of operator's console.

(b) Lamp Test: Test the lamp connections.

(c) Buzzer Reset: Reset the alarm buzzer.

(3) Data Identification:

(a) Group Name: Specify the data group.

(b) Point No.: Specify the data point identification no. within the group.

(c) Data Type: Specify data type.

(4) Data: Value of data to correspond to identification.

(5) Request Function: Request the system to execute specific task or program, for example,

Cyclic Scan
Scanning Start/Stop
Monitoring Start/Stop

Trend Recording/Logging

Reporting

Operator's Guidance Calculation

(6) Loop Functions: Loop status change or set functions for a control loop of DDC or SCC, for example,

Loop Close/Open

Cascade Close/Open

Guidance

Manual

4.3 Notations

4.3.1 Group Name

The group name is represented using numeric or alphanumeric characters.

4.3.2 Point No.

The point no. is represented using one alphabetic character and three or four numeric characters. This code is 4-bits code or ISO code, and the alphabetic character's meaning is shown in Table I.

TABLE I

ALPHABETIC CHARACTER IN POINT NO.

Alphabetic	Meaning	Code
X	Miscellaneous (1)	0000
F	Flow	0001
T	Temperature	0010
P	Pressure	0011
L	Level	0100
A	Component	0101
D	Density	0110
S	Speed/Rate	0111
W	Weight	1000
Q	Heat Duty	1001
V	Viscosity or Voltage	1010
N	Miscellaneous (2)	1011
U	Miscellaneous (3)	1100

4.3.3 Data Type

Data types in standard operator's console are generally selected to a maximum of 16 kinds from Table II.

4.3.4 Engineering Units

Engineering units which are displaced are generally selected to a maximum of 12 kinds from Table III.

5. DATA FORMAT

5.1 General

Data formats in the standard operator's console are divided into two classes. One is the data identification part, and the other is the data value itself.

5.2 Data Identification

(a) Format

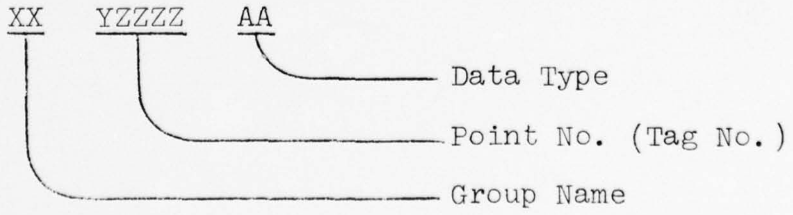


TABLE II

TYPICAL EXAMPLES OF DATA TYPES

Abbreviation	Meaning
PV	Process Variable
SV	Set Point Variable
MV	Manipulated Variable
PH	Process Variable High Limit
PL	Process Variable Low Limit
AV	Averaged Value
SM	Summation
P	
I	
D	
ΔT	Sampling Time
DV	Deviation
SH	Set Point High Limit
SL	Set Point Low Limit
MH	Manipulated Variable High Limit
ML	Manipulated Variable Low Limit
SS	Scale Factor Span
SB	Scale Factor Bias
FT	Filtering Time Constant
RV	Raw Variable
CV	Calculated Variable
HH	Process Variable High High Limit
LL	Process Variable Low Low Limit
ON	ON
OF	OFF
ST	START
SP	STOP
XX	Miscellaneous

(b) Group Name

Group Name is expressed by using two alphanumeric characters.

(c) Point No.

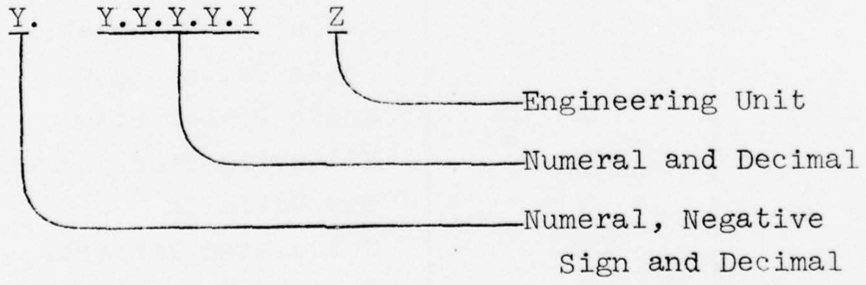
Point No. is expressed by using one alphanumeric character and three or four numeric characters.

(d) Data Type

Data type is expressed by using two alphanumeric characters.

5.3 Data Value

(a) Format



(b) Value

Value is expressed as a moving decimal point type and consists of 6 figures for a positive value or 5 figures for a negative value.

TABLE III

TYPICAL EXAMPLES OF ENGINEERING UNITS

Notation	Displayed
°C	DEGC
%	% or PCT
m	M
ton	TON
kl	KL
kW	KW
m/sec	M/S
t/hr	T/H
Kg/hr	KG/H
Kl/hr	KL/H
m^3/hr	M3/HR
Nm^3/hr	NM/H
Kg/cm^2	KGSC
mmHg	MMHG
ppm	PPM
kcal	KCAL
V	V
A	A
kwh	KWH

(c) Engineering Unit

Engineering units are expressed by using four alphanumeric characters or special notation.

Takashi Tohyama

Chiyoda Chemical Engineering
and Construction Company Ltd.

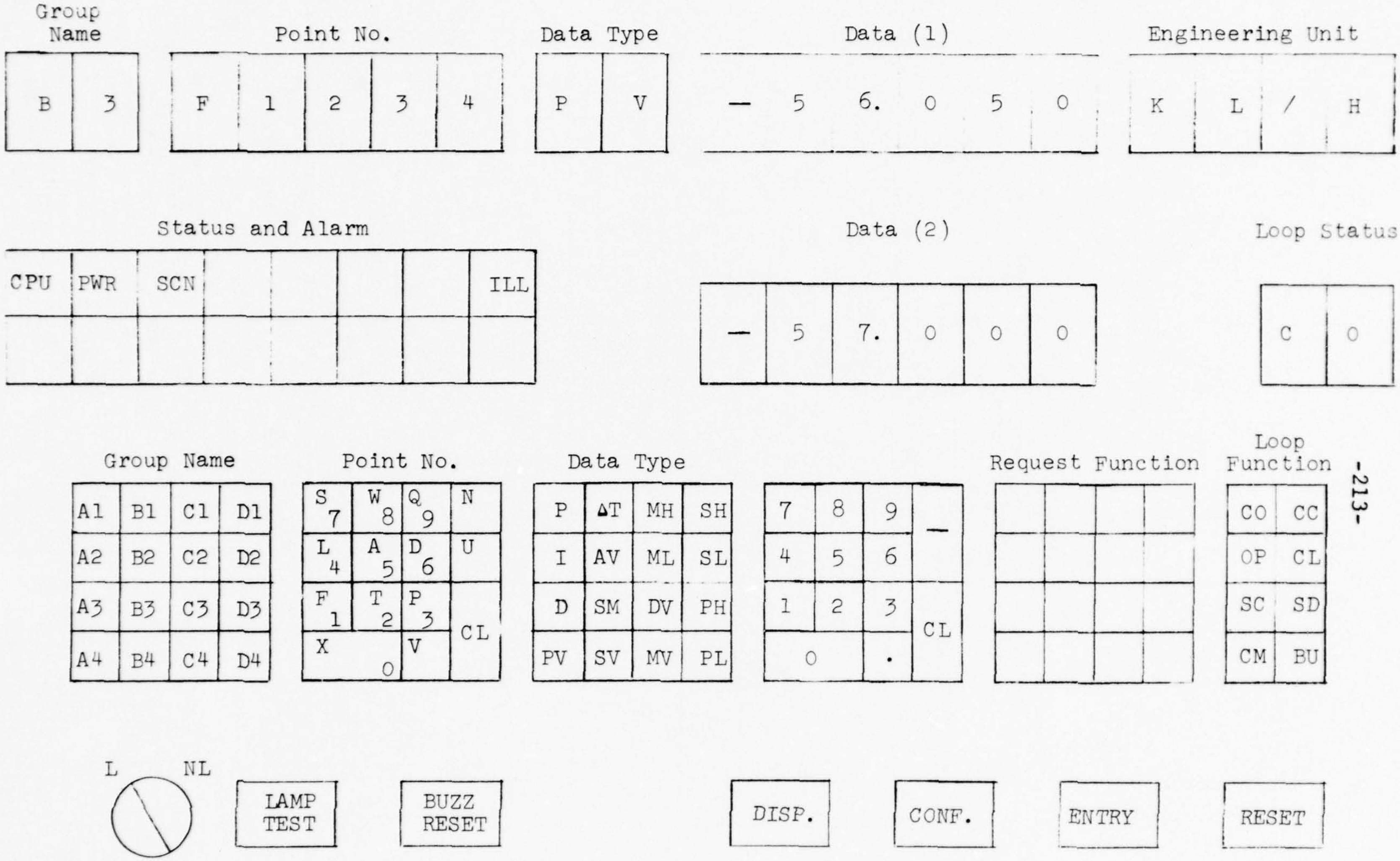


FIGURE 2
NUMERICAL KEYBOARD

Future Operator Consoles for Improved Decision-making and Safety

R. DALLIMONTI, Honeywell Inc.

Computer and crt technology have now reached a stage which makes the "control-room-on-a-desk" *a practical design for large continuous process units*. The flexibility of this man/machine interface permits us to view it as the long sought "adaptive control center." The obstacles to its widespread adoption will be neither cost nor technology. The constraints *will* be our understanding of the operator's job, his operating procedures, and the rate at which new approaches can be absorbed by people.

THE CHEMICAL PROCESS INDUSTRIES (CPI) are undergoing trends in process design, control system strategy, and operations reorganization that call for innovative reappraisals of control room interfaces and a long range view of the functions of process operators. Both the quality and the safety of plant performance are at the heart of these considerations. Modern computer and display technology will provide powerful new answers to these man/machine interface problems of the 70's. Industry is, at last, ready for the long hypothesized "desk-top" control room; moreover, the hardware and economics are now adequate to justify its implementation. The real hurdle will be acceptance of such a radically changed operator interface.

Such a development would unquestionably open up a whole new vista for safety practices in plant operations. To anticipate what these might be, let us first review the most significant trends in modern plant design and operation that impact on safety and which will influence the future design of operator interfaces in control rooms. The list is familiar, but it helps to review if only to ensure that there really are improved answers for each factor:

1. Single train, in-line units with minimum backup equipment
2. Larger units with higher throughput rates and consequently higher stored energy systems
3. Greater interaction between units, resulting from increased integration of energy recovery systems
4. Faster dynamics resulting from reduced intermediate storage and unit buffering
5. Increased centralization of control into fewer and larger control rooms, resulting in higher instrument density per operator
6. Increased use of electronic control systems
7. Continued growth of computer-based operations
8. More complex and integrated control strategies aimed at operation closer to process and equipment constraints
9. More on-line process improvement investigations, made possible by more flexible computer control systems, which may increase risks
10. Increasing demands on operator skills and know-how

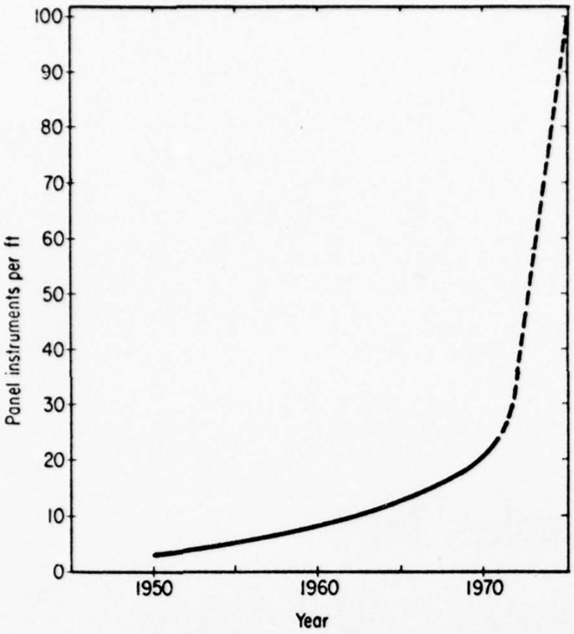


Figure 1. The trend in data density on control panels has been rising, but appropriate computer interfaces could produce a fourfold increase in the next five years.

11. More frequent changes in process design and operating practices, making it difficult to maintain operator know-how at desired levels

12. Increased multiplexing of equipment to reduce capital investments

13. Stronger public and governmental pressures to reduce industrial pollution

14. New governmental regulations for ensuring the safety of workers in process environments.

These particular trends have been isolated because they ultimately intersect at the man/machine interface in the control room. All of the factors listed above stress more than ever the need for a control center where information may be concentrated, made available quickly in a form permitting rapid decision-making, and with efficient means for manipulating controls.

A new interface needed

The CPI have been outstanding in innovations at the operator interface. Figure 1 indicates the display density trend that has been experienced in control rooms. Remarkable as it may seem, this decade *can* see information density take off at the rate shown by the dotted line projection.

The means of this increase is the desk-sized console, at which an operator has access to all data display and control functions now provided through conventional control panels. In spite of miniaturization and centralization, these functions are now still largely performed through an interface that can stretch over 10 to 40 feet of instrument panel per operator. Furthermore, computer control, as implemented today, requires monitoring of additional areas in the form of operator consoles and various printout devices. However, the latter displays are rarely physically convenient to the extended panel areas. From a human engineering viewpoint, it is difficult to reconcile these rather disparate interfaces.

The future goal seems clear – a single interface physically accessible to a seated operator. The data and controls should come to the man and not the reverse. Just a few years ago the idea of running major process units from a desk-sized console would have been considered too "blue sky," but this possibility can no longer be taken lightly.

Establishing the specs

Can we develop the basic requirements and show the viability of such an exciting prospect? In particular, in view of our concern for safety, how can it be made to further the cause of safe operations? Let us first identify those aspects of operator performance where considerations of safety enter.

- Continual monitoring of key-operating variables for deviation from the norm
- Monitoring for malfunction of process equipment or control system elements
- Detection and interpretation of alarms
- Prompt access, display, and control of pertinent data during upset conditions
- Proper implementation of emergency procedures
- Proper implementation of startup and shutdown procedures
- Special surveillance of systems affected by current maintenance operations
- Direction and guidance of field operators during equipment switchover
- On-line ability to check system calibration and performance

Most of these requirements are fairly obvious. Are there more subtle aspects of the operator interface that should be appreciated as we plan the design of the desk-top control room?

Field study of operations

Starting several years ago, we could see the approaching technical and economic feasibility of this new concept. We at Honeywell decided that an updated picture of control room practice was essential before launching off in such a radical direction. We embarked on a program of direct and extended observations of operators and supervisors in action. This was not to be just a polling of opinions or a collection of speculations, but as much as possible a gathering of quantitative measures of operator actions in live control room environments. To do this, of course, required the cooperation of industry, and we were extremely gratified by the response of so many companies who allowed us to live in their control rooms and associate with operators and supervisors for weeks at a time on all shifts.

We had a unique opportunity to observe and measure those many aspects of plant operation that exist in the central control rooms of large continuous processes. We noted techniques used by operators to monitor displays; we counted the nature and frequency of manipulative actions; we studied the use to which recorder information was put. In short, we tried to gather, as quantitatively as possible, data that would be useful in an engineering assessment of new approaches to the man/machine interface in control rooms of the future. In the course of this study we have held many candid discussions with dozens of operators, foremen, and unit supervisors.

The study explored operating interface problems in about a dozen U.S. installations representing processes from gas plants, to ammonia, to integrated refineries, to olefins petrochemical complexes. These were relatively new installations, most being no older than three years. About half involved some form of computer interface.

A summary of the observations most pertinent to future console designs is:

1. A universal shortcoming of all control centers is the rigidity of display which results when instruments must be mounted firmly in place at a fixed position in a panel. All systems undergo continual change, and present day interfaces are very difficult to modify to keep up with the changes. What is needed is a form of "adaptive" interface – now technically feasible.

2. Operation by exception is pretty much the way operators monitor, whether they consciously recognize it or not. The deviating red pointer (or equivalent) was used significantly by 90 percent of all operators interviewed. It provides that first quick-look appraisal of overall plant status. However, while practically all operators endorsed the use of deviation indication, at least 50 percent of the operators interviewed preferred trend recorders for exception monitoring.

3. At the first level of process monitoring, operators do not use quantitative information, if some form of analog displ.y exists. That is, they establish certain visual patterns from the displays which they associate with good operation, and they look for this. This might be called operation by graphic pattern recognition.

4. When quantitative information is needed, an operator will favor digital displays. If both are available, he will prefer the digital form as long as it is conveniently accessible to him in a physical sense. That is, if he is at a panel where an analog display exists, he will not move to a console just to get the same data in digital form.

5. Operators require grouped information for more effective diagnosis and prediction.

6. Each state of plant operation has a preferred set of key variables that are most convenient to scan.

7. Alarm systems in general are unsatisfactory, particularly those in computer systems which rely on typewriter printout. Alarms will mushroom after a system is installed, and better alarm hierarchy strategy is needed. Everyone shudders at the analysis job required to plan and rationalize such systems.

8. Most present computer consoles are not satisfactory for the typical operator. They seem designed more from an engineer's viewpoint than from an operator's. Primarily, the complaint is against the complexity of procedures for accessing and inserting data. When given a choice, most operators want dedicated function pushbuttons rather than touch-tone, coded entry keyboards. Consequently, ' hen split interfaces, such as an instrument panel and a console exist, many operators will favor the panel as the source of data.

9. Operators can generally do a better job of quick recovery from most disturbances by going to manual control, even when the automatic controls would have done the job adequately. In some cases, the controls cannot cope with larger disturbances, and operations must go manual.

10. Graphic panels and other large mimic displays are of questionable value after the initial learning period.

11. Most operators are very adaptable and soon learn to work efficiently, even on poorly designed interfaces and in rather unfavorable environments.

12. The continuing trend to centralized control is resulting in reduced manpower with a resultant increase in the number of supervised loops per operator – and it is working!

13. At least 60 percent of operators studied had the ability to absorb a more sophisticated understanding of operations and to assume broader decision responsibilities in meeting operating objectives than they had been given.

In addition to these more tangible points of observation, one accumulates a broad appreciation of the spectrum of upsets and emergencies that can occur. These are difficult to quantize and are variable from process to process.

Parallel vs serial interfaces

An important consideration in the design of compact control centers is the degree of parallel information display. In the traditional instrument panel, an operator has continuously deployed before him all the control data that are available. Computer consoles have invariably provided serial output of information. This results in a more difficult flow of data to the operator, and hence there is still considerable dependence on the displays of the large panel. The same type of considerations are involved

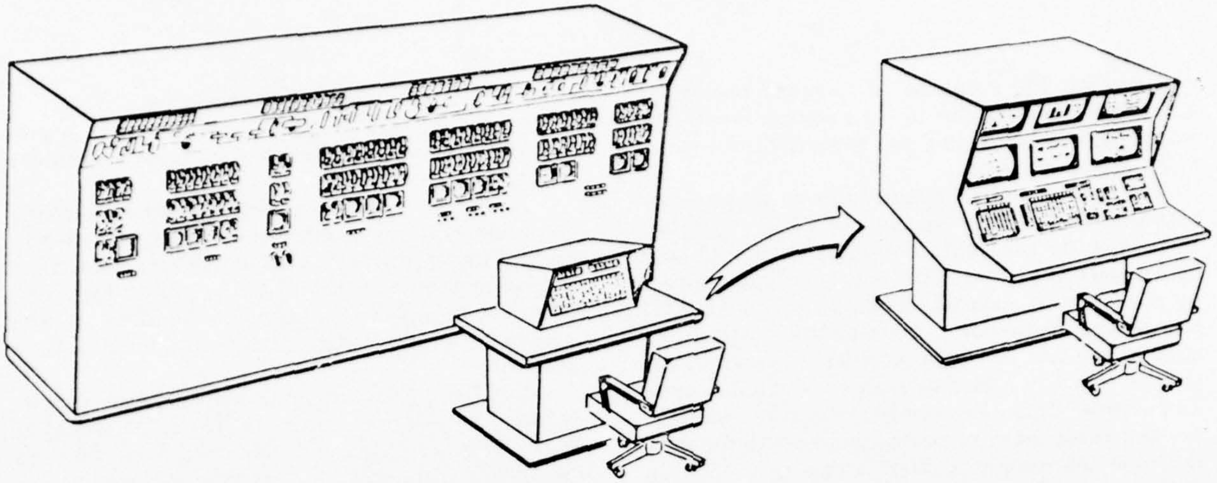


Figure 2. It is technically possible now to condense all an operator needs into a desk-sized control center.

when it comes to supplying manipulative controls. The traditional panel achieves the extreme in dedicated control adjustments. Each loop has its individual set of control adjustments. In present day computer consoles, control adjusting devices are shared among various control loops and various functions. Thus, this whole issue of the degree of parallel or serial function flow is crucial to the design of compact control centers.

It is difficult to converge on a rigorous answer to this problem. A considerable amount of exploratory work and experience must be accumulated. Clearly, the typical operator's console designed today for most computer systems is totally inadequate as a single interface for most processes. Primarily this is due to the extreme sequentialness with which one must operate. On the other hand, from a human factors consideration, it is clear that the human operator can only see one display and manipulate one control at a time, even when he is given the fully parallel facility of the traditional instrument panel. One suspects that there must be some optimum combination which will produce the most effective control center. As far as data presentation is concerned, it need not go to the extreme of a continuous display device for each variable, but neither can it share a single display device across all variables.

As for manipulation of controls, there is room to argue that perhaps a single manipulator is shareable across all loops requiring adjustment. Careful examination of operators during emergency conditions in control rooms strongly supports this contention. One should not confuse the presence of two men at a control panel with the essential need to make multiple adjustments simultaneously. The vast majority of processes are controllable by rapid *sequential* adjustments as long as information feedback is presented, properly organized, and within easily observable distances. Usually several operators are required during upsets merely because physical dimensions of the interface preclude convenient and rapid adjustment by one man. In other words it is a requirement imposed by the interface and not by the process!

Interactive crt consoles

As a result of such studies and much agonizing over the uncertainties of plant performance and operating practices, the author is convinced that we will have technically sound solutions to practically all of these issues within the time frame 1975 to 1980. It will be achieved by the maximum exploitation of interactive, graphic crt consoles in computer-based process control systems. The technology is all in place today. The costs of implementation are coming down.

The control center is on the threshold of another major shrinkage. The panel that now runs 10 to 40 feet can be the sleek console center illustrated in Figure 2. Yet dramatic as the size comparison appears, the real gain is in the more effective interface through which the operator will work.

The transition from the control rooms of 1970 to those feasible by 1980 is illustrated in Figure 3 by the block diagrams of operator interfaces. Figure 3A shows a typical layout of conventional computer control systems. Characteristically, there is some kind of instrument panel on which are mounted either ddc or setpoint control stations and alarm annunciators, a separate console which permits communication between operator and process via the computer, and a variety of printout or logging devices. As was pointed out earlier, the operator must work between two information centers, neither one being totally adequate to stand alone in all situations.

However, with the advent of the interactive crt console the system will look something like Figure 3B. All information and manipulation is now within arm's reach of a seated operator. Both computer operation and backup system operation are possible from the same console.

The extreme flexibility of design made possible by a display device which can draw almost any desired picture, table or graph, coupled with the convenience of compact, dedicated function keyboards, is a resource for tremendous innovation. The mechanics of interfacing a computer with various crt's and a functional keyboard are already well established, so the real challenge in this concept is the structuring of information displays and data access procedures for maximum usefulness to operators. Already there exist numerous examples of the use of this technique in process control (Ref. 1,2,3). However, in no case known to the author has this approach been carried to the point of being the sole interface to the process. This is now a realistic objective.

The ideal console should provide a basic set of functions which would make it at least the equivalent in operability to the traditional long instrument panel. High on the list would be a display that conveys the deviation of all points under manual or automatic control. This should be accomplished without the need for the operator to dial in codes or call for each point to be examined. It has been demonstrated that deviations for as many as 100 loops can be displayed simultaneously on a single crt face (Ref. 4). Thus one gets the overview pattern for quick monitoring of the total process state. It is readily appreciated that other techniques of pattern recognition can be evolved.

Next, it should be extremely easy to select any loop for detailed examination and manipulation of controls. Many techniques are possible and have been proposed. The most intriguing one is that which permits the operator to touch the image with his finger at the point requiring more detail and having that detail subsequently appear. This is being done. Other methods are described in the literature (Ref. 3,4).

All points previously on recorders can still be displayed, on call, utilizing memory stores for preserving the data (Ref. 2).

The use of interactive graphic displays should minimize the need to punch in long code strings of alphanumeric data via keyboards.

New techniques for communicating alarms are opened up. It is now feasible to display several hundred alarm indicators in the space of an 8-1/2 by 11 in. page. Alarm messages and diagnostic aids can be flashed on a scope in a matter of seconds. By compressing the overall alarm picture into an area that the eye can scan without even a head movement, the time to detect and respond is reduced. It becomes possible to evolve techniques of diagnosis based on graphic arrays: another aspect of process monitoring by pattern recognition.

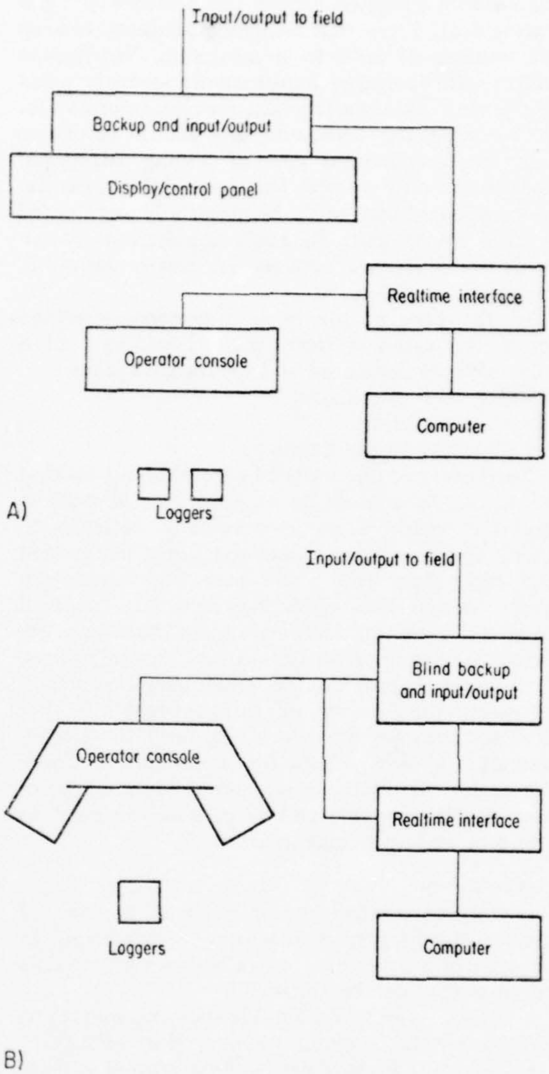


Figure 3. Present system layouts (A) have the computer tacked onto a complete conventional system. Controls built around a computer will look more like (B), with all data requirements directly before the operator.

Note that with the condensation in area of this interface some of our old ideas about display specifications must change. For example, it is very common for specs to call for indicators that can be read from across the room. Is this requirement necessary once we can bring all the essential operating information into an area the size of a desk top? Another item: control room lighting requirements will probably change to suit this form of presentation. Better control of panel glare will be possible.

Pitfalls in implementation

There are a number of temptations that will lure the console designer. Under the pressure of minimizing cost, there will be strong tendency to keep the number of crt's to a minimum. But human factors and operating requirements strongly point to the need for simultaneous, parallel information. In assessing the trade-offs, one should recognize that the incremental cost of adding additional monitors is only several thousand dollars, yet the interface capability could be materially augmented by this investment. In some applications it can mean the difference between success or failure in operating feasibility.

In the case of the basic functions described above, the author believes there should be a minimum of three dedicated and separate displays:

- Overall system monitor
- Alarm monitor
- General-purpose monitor.

Furthermore, one must be careful about making the operator's console be all things to all people. Again, it would seem economically desirable to satisfy the needs of process engineers, instrument engineers, supervision, and even higher management through the same interface. The concept lends itself to this end; but again, based on observed human interactions in real control rooms, this may not provide the most useful solution. However, the beauty of this approach is that auxiliary consoles are feasible to serve these other functional groups within the organization. These can tie into the same data communication base of the operating console and may even be viewed as a form of backup in emergencies.

Aids to safety

The desk-top control center becomes a powerful tool for exploring new techniques of operation. In the area of safety there are a number of obvious functions that can be enhanced:

- Alarms – Immediate guidelines and messages, diagnostic aids, procedures conveyed by pictures
- Startup and shutdown – Procedure checklists dynamically updated and sequenced
- Maintenance – Summary displays of equipment currently under maintenance and readiness status
- Contingency predictions – Prediction of possible future process states based on present trend picture, prediction of hazardous conditions
- Training – During periods of quiet operation, on-the-job programmed teaching can be carried out to upgrade and refresh operator skills. Training by simulation is a real possibility.
- Closed-circuit TV – Portable or fixed cameras in the plant can monitor progress of emergency or maintenance operations. These can be mixed into background of data displays. Routine scanning of running equipment can be done from control room.

These are just a sampling of possibilities. The impact on operations safety can only dimly be seen at this time, but certainly the potential is high.

At what price?

Finally, the big question: How much? The typical answer is, "it depends." We assume that a computer has already been justified. Then one can add any one of 50 commercially available crt terminals for as little as $5,000 for the hardware. Accompanying this cost, there will also be a programming cost that could be as little as a man-week to open-ended at the other extreme. Of course, this would not implement the total "control-room-on-a-desk" concept: that cost is certainly more controversial and difficult to assess at this time. Based on "ball-park" cost studies, it seems feasible to assemble, for about $50,000, the hardware for a console performing the basic functions defined earlier and serving the needs of a 100-loop control system. In judging these economics, keep in mind that the costs of traditional instrument panel display have been eliminated. Of course, there is still the equivalent cost of backup control and other process I/O buffers.

At this time, the net cost trade-off may be from break-even to 15 percent higher for the new desk-top console. However, there have been others who claim rather drastic reductions in cost by as much as 50 percent (Ref. 3).

As with most computer interfaces, there can be a substantial software investment required to program the more elaborate interactive systems that are possible. This area is hard to estimate, since it is so dependent on the scope and flexibility of the display philosophy. It is not the intent here to arrive at a rigorous cost prediction but rather to suggest strongly that cost itself will not be a real deterrent to the implementation of this concept. There is every prospect that the decreasing cost trend of this display equipment will continue at a greater rate than that of more conventional interfaces. Thus, the economic picture will almost certainly improve.

References

1. Aronson, R. L., "CRT Terminals Make Versatile Control Computer Interface," *Control Engineering*, April 1970, p. 66.
2. Crowder, R. S., "CRT Interfaces for a Continuous Plant," *Instrumentation Technology*, January 1971, p. 58.
3. Morris, A. H., *et al.*, "Are Process Control Rooms Obsolete?", *Control Engineering*, July 1971, p. 42.
4. Lauher, V. A., *et al.*, "CRT Control Center for a Multiloop Process," *Instrumentation Technology*, September 1970, p. 33.

RENZO DALLIMONTI *is a member of the Advanced* Technology Staff, Honeywell Industrial Div., Ft. Washington, Pa. This article is based on his paper presented at the 13th ISA Chemical & Petroleum Instrumentation Symposium, 1972, Philadelphia.

SECTION III

GUIDELINES AND OTHER DOCUMENTS OF THE SYSTEM RELIABILITY, SAFETY AND SECURITY COMMITTEE

The attached documents give an excellent reading of the work of this Committee. They include:

1. "Assurance of Operation of Industrial Process Control Systems", *Minutes of the Second Purdue Meeting of the ISA Computer Control Workshop*, Appendix V, pp. 170-204, Reprinted from Technical Committee 65, International Electrotechnical Commission.

2. "Loss Prevention Guidelines for Process Control Equipment", *Ibid*, Appendix II, pp. 119-129, by Thomas M. Riley.

3. "Working Papers of the European Branch, System Reliability, Safety and Security Committee", *Minutes Third Annual Meeting International Purdue Workshop on Industrial Computer Systems*, pp. 345-446 as follows:

   a) "Application and Functional Test of Self Checking Programs: Their Influence on the Failure Probability of Computerized Safety Systems", by H. Schuller.

   b) "Safe Computer Systems Hardware - Part 1", by H. Schuller and W. Schävier.

   c) "Remarks to Revision of Methods to Develop Safe Computer Systems", by H. Trauboth.

   d) "Computer Safety and Security - Back to Basics", by J. R. Ellison.

   e) "Methods to Develop Safe Computer Systems, by H. Trauboth.

f) "Safe Software by Functional Diversity", by R. Lauber.

4. "The Guideline for Safety of the Industrial Computer Systems", a new contribution of the Japanese Branch of The Committee to be published in the next Minutes.

ASSURANCE OF OPERATION OF INDUSTRIAL PROCESS CONTROL SYSTEMS

INTERNATIONAL ELECTROTECHNICAL COMMISSION

Technical Committee No. 65

Industrial Process Measurement and Control

The present document has been established by the Working Group 3 after the meeting in Venice on April 13th and 14th 1972 and in view of the comments of the National Committees.

TABLE OF CONTENTS

ASSURANCE OF OPERATION OF INDUSTRIAL PROCESS CONTROL SYSTEMS

1. GENERAL

1.1. Introduction

Control systems are increasingly replacing human effort and skill in the operation of industrial processes. These systems which can, for example, be made up of measurement and control equipment, instrumentation, and computers, may be intended to control the whole or part of the operation of a process, or more simply to monitor a particular process variable or function. The ideal system is one which achieves two basic requirements. The first requirement is that the system should fulfill the specified performance in the given environmental conditions. Secondly, the system should keep this performance throughout its whole operational life without failure or degradation.

In fact, such an ideal system will never be achieved. Equipment faults can never be totally avoided even though technical improvements in components, better design of system configuration, more stringent methods of testing and rigorous preventive maintenance procedures are continually being developed.

A knowledge of the reliability of control system elements by calculation, testing or estimation may give a means of selecting the most efficient system in respect of the frequency of failures during the operating time of the system. Nevertheless, it is necessary to go further in many cases and take into account, not only the frequency of failures, but also the consequences of the failure on the controlled process. If the consequences of a control system failure due to an equipment fault or human error are analyzed, it will be seen that some failures can produce conditions in the process which may be hazardous to personnel, the environment, and to the process and its control equipment. Other failures may affect the proper operation of the process so that efficiency is partially or completely lost, and the desired end product is not obtained. Each failure will produce its own level of danger or loss of efficiency. The avoidance of these undesirable system failures and the protection of the process and its environment will dictate many aspects of the control strategy and will influence equipment design features, manufacturing, installation, and testing procedures; and, of course, the cost of purchasing and operating the system. In such a case it can be said that the control system has been designed and installed according to a particular degree of Assurance of Operation.

1.2. Object

The object of this document is, in Part A, to introduce the concept of Assurance of Operation, to present methods of analysis which will allow the idea to be specified in objective terms, and to provide guidance to control system users and manufacturers in compiling and verifying the sections of a process control system specification which apply to operational assurance. In Part B, the document presents a number of guidelines to good practice which will assist manufacturers and users in schieving the desired Assurance of Operation in their systems.

2. DEFINITIONS (To Be Completed)

System

Process

Control

Fault

Failure

Human Error

Control Action

Monitoring

Reliability

Availability

Maintainability

Safety

Hazard

Safeguard

Assurance of Operation

System Quality

Fail Safe

PART A: Assurance of Operation

A.1. Introduction

Assurance of Operation of a process control system is intended to qualify the system as far as reliability, availability, maintainability and safety are concerned. It is recognized that the probability of failure of a system is not, by itself, a complete measure of system behavior in time. Instead, the definition of a failure of the performance of a system is stated by combining the probability of occurrence of a fault with an analysis of the consequence.

In one case the only consequence of a control system failure may be that the process stops for two days until the maintenance engineers can find and rectify the fault. Alternatively in another case, a particular fault sets up a chain reaction in the process which within 10 minutes would cause a large explosion, probably fatally injuring the operating staff and wrecking the plant. In most cases the safety of personnel and environment are of paramount importance, but two days total downtime may also be important when considering the economic viability of a process.

Taking into account the four most important consequences of a fault, it is possible to describe Assurance of Operation as the probability of occurrence of a specified control system failure in a certain time period, weighted in relation to the hazard the fault causes to personnel and environment, to the potential severity of injuries to personnel and plant, and to the loss of revenue from the process.

By combining these four variables, a measure of the true impact of a fault could be obtained. If each variable contributing to Assurance of Operation can be quantified in meaningful terms, it is then possible to specify Assurance of Operation in objective terminology, and as such, it can be incorporated in a system specification in the same manner as system performance, environmental limits, etc. The quantitative analysis can also be used as a tool for determining compliance of the equipment against the Assurance of Operation specification.

A specification of Assurance of Operation would take the form of quantitative definitions of the variables which contribute to Assurance of Operation, i.e., Probability of Occurrence, Severity, Hazard, and Economic Loss, followed by limits for the four variables which must not be exceeded for any conceivable equipment fault.

In the following paragraphs the variables of Assurance of Operation are described in more detail and a set of simple classifications is suggested.

A.2 Failure Probability

The probability of occurrence of any particular identified fault can be estimated by means of a reliability analysis of the system element that has been assumed to have had failed, and can be expressed in quantitative terms. This probability provides a measure of the expected number of occurrences of each identified failure cause, during the specified period of equipment life being considered.

If quantitative failure rate data is available, the failure probability for individual faults can then be expressed in the number of failures over the operate time (failure rate λi; multiplied by time $ti = \lambda t \cdot ti$). This value is then used to establish the location of this particular fault on the probability of occurrence axis of the four dimensional Assurance of Operation space.

Because of the great variety of probability values, the analysis becomes more meaningful when faults are grouped into logical pre-established ranks that reflect the complexity and performance of the overall control system.

(a) Example of grouping of faults by probability ranks, e.g.:

Probability Rank 1 - Any fault with $\lambda i.ti$ smaller than 0.02

Probability Rank 2 - Any fault with $\lambda i.ti$ between 0.02 and 0.04

Probability Rank 3 - Any fault with $\lambda i.ti$ between 0.04 and 0.06

Probability Rank 4 - Any fault with $\lambda i.ti$ between 0.06 and 0.08

(b) Example of Grouping of faults by contribution of of system probability

Such grouping relates each fault to the assessed overall fault probability of the system ($\lambda s.ts$) rather than letting each absolute fault probability ($\lambda i.ti$) stand on its own. This relationship is established by the ratio of the individual fault probability ($\lambda i.ti$) to the overall equipment fault probability ($\lambda s.ts$). e.g.:

Probability Rank 1 - Any fault that contributes less than 2%

$\left(\frac{\lambda i.ti}{\lambda s.ts} = 0.02\right)$ to system fault rate

Probability Rank 2 - Any fault that contributes between 2% and 4% to system fault rate

Probability Rank 3 - Any fault that contributes between 4% and 6% to system fault rate

Probability Rank 4 - Any fault that contributes between 6% and 8% to system fault rate

If quantitative failure rate data is not available or is suspect, relative probabilities of individual faults must be established, based on engineering judgments and prior experience.

To facilitate a consistent and traceable way of recording such judgments, the methods below are suggested.

(a) Fault probability grouped by frequency of occurrence, e.g.:

Probability Rank 1 - Any fault that occurs less than one time in one year

Probability Rank 2 - Any fault that occurs more than one time in one year, but less than 2 times

etc.

(b) Fault probabilities grouped by contribution to system failure probabilities, e.g.:

Probability Rank 1 - Fault occurrence very low (less than 2% of all faults)

Probability Rank 2 - Fault occurrence low (2% to 4% of all faults)

etc.

Classification Table for Probability of Occurrence

Rank	Probability $\lambda i.ti$	Contribution $\frac{\lambda i.ti}{\lambda s.ts}$	Frequence of Occurrence	Per Cent of Contribution
1	0.01	0.01	0.5 X/Year	1
	0.02	0.02	1.0 X/Year	2
2	0.02	0.02	1.0 X/Year	2
	0.04	0.04	2.0 X/Year	4
3	0.04	0.04	2.0 X/Year	4
	0.06	0.06	3.0 X/Year	6
4	0.06	0.06	3.0 X/Year	6
	0.08	0.08	4.0 X/Year	8

Fault Rate Data Available

Fault Rate Data Sparse

Fault rate of element (i) = λi

Operate interval of element (i) = ti

Fault probability of element = $\lambda i.ti$

Fault probability of system = $\lambda s.ts$

A.3. Level of Severity

For each individual fault the effect can be translated into ranks of injury potential, identified as levels of severity, so that this information becomes one of the scales for measuring Assurance of Operation. The definition of the ranks should reflect the application and environment of the control system being examined. An example of the level of severity ranks, and broadly applicable definitions, is given below.

Classification of Levels of Severity

Rank	Level of Severity (Injury Potential)
1	Fault will not result in personnel injury.
2	Fault will cause minor injury, e.g., minor cuts, bruises.
3	Fault will cause major disabling injuries.
4	Fault will cause extremely serious injury, e.g., amputations, permanent disability.
5	Fault will cause fatalities.
6	Fault will cause a catastrophe, numerous fatalities.

A.4. Level of Hazard

The degree of hazard generated by a fault is related to the time (T) that is available to implement corrective action to avoid injury. The less time there is to mitigate injury, the higher the Level of Hazard. The fault that causes injury without warning, where the time available for taking action approaches zero, is identified as "1.0," the highest Level of Hazard. At the other end of the spectrum is the fault that is of such a nature that correction is not necessary to avoid injury, and safe operation is maintained throughout the remaining life of the equipment, without corrective action. This fault is classified as "0," the lowest Level of Hazard. The Level of Hazard index which is a measure of the degree of the safety hazard is expressed as e^{-T}, where T is the actual time available to implement corrective action to avoid injury.

Classification of Level of Hazard

Rank	Hazard Index, e^{-T}
1	0 0.25
2	0.25 0.50
3	0.50 0.75
4	0.75 1.00

The determination of the actual available time requires the evaluation and analysis of the following time intervals.

(a) Tc - Time available to implement corrective action (time interval from occurrence of fault until injury occurs).

(b) TR - Time required to recognize the existence or presence of a fault condition, or the time required for an automatic safety device to react in the presence of a failure to prevent injury from such a fault.

(c) T - Actual time available to take action T = Tc - Tr.

Determining the available time, T, also depends on the presence (or absence) of built-in instrumentation or monitoring devices. The time required to recognize a fault condition is less if it is indicated by a warning device. The hazard in such a case is less than if detection of the fault condition is left to the experience or alertness of an operator.

PART B: Guidelines to Good Practice

B.1. Introduction

The reliability and operational integrity of an industrial process control system is determined by a number of highly interdependent factors:

1. Nature of the process to be controlled
2. Performance requirements of the control system
3. Control system reliability and maintainability
4. Physical and other environmental requirements
5. Required delivery schedules
6. Total installed cost of the control system

These factors must be considered in varying degrees, depending on the intended end use of the system. Any successful system represents a compromise between them.

A key criterion in the realization of a successful control system is the understanding developed between the supplier and user regarding the meaning and significance of various control system characteristics. There are numerous points satisfying each of the above factors which must be considered by both supplier and user. Many of these may be easily forgotten or glossed over, resulting in a less than satisfactory control system.

Some of the key questions which must be answered by the supplier and manufacturer are:

1. Has the user properly analyzed his process to determine the control strategies, human interface characteristics, etc., required?

2. Has the user specified the proper control system for his process?

3. Has the system manufacturer correctly interpreted the customer's specifications?

4. Has the manufacturer designed the control system so that in the event of element malfunction the system will shut down or the final elements will travel to position known to be safe?

5. Are the elements selected of such reliability so that their failure rates are properly related to the expected servicing and checking intervals?

6. Are the installation, maintenance, and servicing instructions explicit and complete enough to keep the system operational over the required periods?

7. Have damages to personnel and equipment been properly identified, and have the proper warning notices been posted and safety precautions taken?

It is the purpose of this section to provide a check list of guidelines to be considered during the evolution of the control system, to assure that the pertinent points have been considered. Many of these points do not apply to every control system; however, the important point is to assure that they have been considered and found not applicable, rather than forgotten.

B.2. Process Requirements and Specifications

We are concerned with the specifications generated by customers(as designers, in fact, of the overall process control system) and required by manufacturers in the context of the "Assurance of Operation of the Control System."

This clearly is only a portion of a total specification, but the specification relating to Assurance of Operation must embrace the following aspects:

Design

Manufacturing

Test

Maintenance

B.2.1. Design Specification

The specification relating to design in respect of Assurance should cover the following aspects:

a. Specification of:

1) Reliability

2) Availability

3) Safety (plant and personnel)

of constituent portions of the control system. In order to achieve this, the specifier must have considered the overall process system. He will have considered the effect of various fault situations and thus will be in a position to clearly define the conditions which constitute a system failure situation. This will enable the control system designer to define areas where redundant elements will be required, and areas where "fail safe" techniques must be applied, in context of plant hazards.

Implicit in any stated availability requirements are aspects of the specification relating to maintenance. These will be discussed under these headings.

Other aspects relating to design to achieve availability:

b. Environmental conditions

1) Climatic - temperature, humidity, dust, fumes

2) Electrical - signal/noise, incoming power supplies

3) Mechanical - vibration and shock

4) Physical and chemical conditions: radiation, corrosion

c. Design for ease of maintenance - e.g., standardized components, modularity of hardware, adequate test points and built-in system, monitoring, and diagnostic aids.

d. Operation profile

B.2.2. Manufacturing Specification

The specification should contain adequate information on aspects of manufacture related to quality of the product and must contain reference to any appropriate standards relating to:

a. Mechanical assembly/finish - codes of practice and standards

b. Assembly and wiring - codes of practice and standards

c. Material and component procurement/quality assurance standards

d. PCB (printed circuit board), etc., assembly and test - codes of practice

e. Unit "typetesting" techniques

The specification should also define any requirements of the customer to visit manufacturing premises during the course of manufacture to examine the Quality Assurance during manufacture.

B.2.3. Test Specification

An essential part of a specification relates to the Testing operation, both within the manufacturer's premises prior to delivery to site, and further, after the site installation and commissioning.

In the context of "Testing for Assurance of Operation," these tests may take the form of extended operation possibly at extremes of temperature together with temperature cycling with some defined "criteria" for measuring success, detailed in an "Acceptance Test Document." This will define equipment failure criteria, methods of measuring equipment repair times, and define a level of spares holding during the test, etc.

B.2.4. Maintenance Specification

Finally, the specification must make reference to the requirements in respect of the maintenance operation.

It must define a spares holding.

It must define requirements for long term availability of spares (if required).

It must define the standard arrangement, form, etc., of the maintenance manual.

It must define the training requirements for staff to maintain the equipment.

The following table summarizes headings of the specification requirements relating to Assurance of Operation.

Aspects of Specifications of Control Systems
Relating to Assurance of Operation

Design Aspects	Manufacturing Aspects	Test Aspects	Maintenance Aspects
1. Definition of Failure Criteria 2. Definition of Reliability, Availability and Safety Requirements 3. Definition of Redundant Element Requirements 4. Definition of "Fail Safe" Area 5. Environmental Specification - Climatic - Electrical - Mechanical 6. Definition of Design Modularity 7. Definition of Diagnostic Requirements	1. Specifications on Standard of Mechanical Assembly 2. Codes of Practice on Assembly and Wiring 3. Quality Assurance on Components and Material 4. "In Plant" Visits	1. Acceptance Test Document Defining "In Plant" and Post Commission Tests	1. Specification of Spares Holding 2. Specification of Maintenance Manual 3. Specification of Staff Training Requirements

B.3. Design of Process Control Systems

Safeguarding of the process forms part of process control. The process is kept in a safe condition by automatic corrective action or by monitoring the process variables and alarm signals.

The following requirements are of particular importance in the design of a process control system:

B.3.1. General Aspects

a. Consideration of limiting conditions from the economical point of view

b. Clear definition of objective or problem

c. Greatest possible simplicity in concept and solution in order to increase reliability, for example

d. Use of separate control functions in order to improve repairability and availability, for example

e. Conduct load analysis (corrosion, wear)

f. Conduct failure analysis (for example, according to MCA--maximum credible accident--)

g. Application of safe-life methods, i.e.,

Worst-case design

Use of reliable structural elements

Use of derating factors

Use of redundant equipment

Use of monitors

h. Application of fail-safe methods (safe condition during failures)

B.3.2. Safeguarding of Process by Means of Control System

The control of the process variables by itself already keeps the process in a safe condition provided that the system is operating normally. The object of the fail-safe behavior is to maintain the safe condition in the event of certain failures in the control system. In some cases it might be necessary to check that the control system itself operates according to the process control specification.

B.3.3. Safeguarding of Process by Means of a Safeguarding Equipment

The control system may include a safeguarding equipment to keep a check on the process variables should these exceed given limiting values. The reliability of this equipment must be determined in the design stage. In many cases the required reliability can only be achieved by application of redundancy in, and monitoring of, the safeguarding system. Application of the fail-safe technique allows the safe condition of the process to be maintained with prior consideration to the failures that may occur in the safeguarding system.

B.3.4. Safeguarding of Process in Emergencies

In certain industrial plants, the possibility of emergencies such as fire, explosion, or escape of deleterious substances cannot be avoided. The damage caused by such emergencies can often be kept within limits if the process control system continues to be operable to such an extent that the process is driven into a safer condition.

During the design of the system, provision must be made for the protection of the control room, of the emergency power supply, and of cables and lines.

B.4. Manufacturing the System Elements

Reliability of a system is closely related to the reliability of the elements which comprise it. Numerous system failures occur because of faults in one or more of the elements. Therefore, the methods, materials, controls, and management used in the manufacturing operations are of supreme importance.

A number of judgement criteria can be established which will increase the probability that the elements are manufactured with the proper reliability considerations. Some of the more important ones follow:

1. Is the step-by-step manufacturing procedure available and documented, and is it being followed?

2. Does the manufacturing procedure include adequate in-process inspection points?

3. How comprehensive is the Quality Assurance Procedure of the final product?

4. What reliability and performance verification has been provided by the manufacturer?

5. Are periodic performance and reliability audits made on the product?

6. How comprehensive and how well controlled is the procedure for making changes to the design?

7. How are design changes documented, and how are these changes conveyed to a customer having an installed system in operation?

8. What records exist describing operation of the element, and the initial design calculations?

9. What design rules or guides exist which will make it mandatory for the vendor's designers to provide reliable designs? (Example: Derating of components)

10. What environmental and operational constraints and specifications are imposed on the element by the vendor?

11. Has a safety analysis been made of the element to determine what the effects of various component failures are?

12. What controls are imposed by the vendor on his suppliers and subcontractors to assure a reliable product?

13. Is the quality assurance function carried out independently of the manufacturing organization, and does it report to a sufficiently high authority in the vendor's organization?

14. What is the procedure and frequency used to assure that all instruments and gages used to calibrate, build, and test the element are kept in proper condition or calibration?

B.5. Assembling the System

Even though all elements may pass the required safety and reliability criteria, the system as a whole may still be unsafe or not adequately safeguard the process. A number of steps should be followed to assure that the system, when installed and commissioned, has adequate provisions to insure the safeguarding.

The most important ones are listed below:

1. Has an adequate understanding been obtained between the manufacturer and user concerning the system safety and operational requirements?

2. Have system tests been formulated and documented to permit clear acceptance--rejection criteria--including simulated process tests?

3. Have adequate requirements been formulated to design and enforce system quality assurance?

4. Does the system have built-in self-checking provisions, and are these in proper order?

5. Does the system require environmental tests, and have they been carried out and passed successfully?

6. Has a safety analysis been made of the system? Is a reliability analysis required?

7. Is the system quality assurance function carried out independently of manufacturing, and does it report to a high enough authority?

8. What provisions exist to assure that the intent of the designer is properly conveyed to the System Assembly Organization?

B.6. Installing and Commissioning the System

Proper System Installation and Commission is a vital part of reliability and safeguarding assurance since improper installation of a properly designed and built system may easily jeopardize the safety or output of the process.

1. Who should be called for help when required--both from within the plant and from outside, such as the manufacturer's representative?

2. Are adequate instructions provided for transporting, packing, unpacking, and installing the system?

3. Are environmental requirements for operation and storage clearly documented and observed? For example, what are the air conditioning or forced ventilation requirements, and are they being followed?

4. What safety codes must be observed? What hazards to personnel and equipment must be guarded against? Are the codes being followed?

5. Is there a listing of all auxiliary equipment required to install and service the equipment--including standard and special instruments, tools, and calibration equipment--and is it available?

6. Are there explicit interconnection diagrams, including terminal numbers, terminal block identification, etc., and is the system connected accordingly?

7. Prior to full system operation, has the system been sectionalized into subsystems, each of which has been separately tested and its performance verified?

8. Has the system been tested for reasonableness before proceeding with full automatic operation--this is, do final elements move in the right direction; are signal orientations correct?

9. Are properly documented procedures available and followed, for coupling the control system to the process--that is, are adjustments made for optimum

dynamic response; has the software system been checked; have the proper disturbances been applied to the system?

10. Have the operators and maintenance men become familiar with all test points, operating switches, and adjustments? Is the purpose of each clear?

11. Have the wiring, piping, shielding and other pertinent requirements been followed correctly?

B.7. Maintenance

It has been shown that maintainability influences the outlay of funds for system acquisition and utilization. This outlay can be minimized by the attainment of several objectives associated with maintainability. These objectives are as follows:

1. Accomplishment of all preventive and corrective tasks in a minimum of time, with the least number of people, and with the minimum restriction of operation on the system.

2. Accomplishment of the tasks with a minimum amount of training of personnel

3. Minimum expenditure of spare parts

4. Least requirements in variety and quantity of tools and test equipment

5. Least support facility requirements

6. Minimum requirement for contractor services

7. Minimum need of documentation

8. Minimum bad influence on reliability

As an indication to the ways in which corrective maintenance operation times can be reduced by designing for good maintainability, the following guide rules are given. The rules can be used as a check list and the appropriate items selected according to the particular characteristics of the system.

B.7.1. Reduction in Fault Location Time

a. Monitoring devices.
The purpose of such devices is to check the operation of the system and/or the conditions of its elements.

b. Devices used for the troubleshooting.
Such devices can be manually or automatically operated. If the devices for monitoring or troubleshooting change, the level of safety or

the availability of the system, due notice of this fact must be given to the user.

c. Marking.

All test points should be marked. The same marks should be indicated on schematic diagrams and layout drawings. Components, elements, connection points, wires and cables should be easily identifiable.

d. Test Points.

Significant test points should be readily accessible and clearly marked. All information required for repair operations, such as the normal values of variables, should be mentioned.

e. Marginal control devices.

f. Possibility of segregation of definite functional units.

g. Logical and consistent arrangement of functional units.

h. Troubleshooting chart

i. Clear, complete, and easy to follow, maintenance documents (including eventual software documentation).

B.7.2. Time Reduction in Repairing Faulty Elements

a. Accessibility.

Component and subassembly mounting design should take into account the possibility of later replacement in accordance with the expected failure rate.

b. Interchangeability.

Similarly identified elements and sub-units should be interchangeable. The replacement of a device by its corresponding spare should only require simple adjustments correctly described in the maintenance documents. Marking of interchangeable elements and coded connectors should prevent errors or accidents.

c. Connections.

Identification of conductor leads and connections should be clear, logical and well-documented.

B.7.3. Time Reduction in Checking and Adjusting Operations

a. Functional adjusting devices.

They should be easily reached and clearly marked; have satisfactory range and resolution.

b. Checking devices.

The instruments needed for checking and adjusting the system should be clearly specified if they are not included in the system.

LOSS PREVENTION GUIDELINES
FOR
PROCESS CONTROL EQUIPMENT

BY
THOMAS M. RILEY
OIL INSURANCE ASSOCIATION
CHICAGO, ILLINOIS

LOSS PREVENTION GUIDELINES FOR PROCESS CONTROL

I. What is the OIA?

A. The Oil Insurance Association or OIA is technically an insurance pool in which some 40 odd different fire insurance companies have pooled their underwriting resources, thereby, providing capacity for large amounts of insurance in a single policy contract.

B. Among our coverages are Property Damage, Business Interruption and Extra Expense. Among the perils are Fire, (including Inherent Explosion), Lightning, Extended Coverage, Vandalism and Malicious Mischief, Sprinkler Leakage and Pressure Rupture.

C. Our Association was formed in 1918 to service the petroleum industry. Basically, we insure refineries and related process', gasoline plants, oil and gas pipelines, and petrochemical plants for the production of ammonia, fertilizer, synthetic rubbers, plastics, and base materials for synthetic fibers.

II. The Function of Insurance Loss Prevention is to:

A. Eliminate the sources.

B. Confine the loss to given area if they cannot be eliminated.

C. Or finally, give a loss occurrence a path of least destruction if the loss cannot be confined.

III. Control Instrumentation in process areas must remain substantially intact for a system to be controlled. Several of the newer processes depend on more critical control to remain within safe operating limits. In addition, many processes cannot safely have a crash shutdown and must be brought down in an orderly manner, some times involving many hours. All these reasons point towards giving the best protection possible to process control systems.

A. The control or data centers are the focal point of the control network. First of all, can the process be controlled from elsewhere? Is the process simple enough to control from scattered locations? Or is the process stable enough to not need constant control adjustments? The control centers for which guide lines are put forth are those in which "No" is the answer for the above questions or where there is a large concentration of value such as a location in which process or data computers are present.

1. Some possible but typical events are:

a. A fire in an adjacent building, process unit, or tank with heat radiating against the center and smoke obscuring the scene. The fire brigade and/or the fire department come and deluge the fire and your center with water, foam, powder or whatever.

b. There is an explosion somewhere in the plant that hurls a chunk of steel 8" thick by 6' by 3' into your computer center.

c. The computer center is downwind of the large spill or rupture with flammable vapors or gasses blowing about the center.

d. A paper, electrical insulation, or electrical fire is burning within the center itself.

(Now these may sound far fetched and improbable but they have occurred. They not only occur, but in the handling of flammable and explosive products, they happen with a certain amount of regularity.)

2. One of the best ways to minimize the hazard to the control center would be to give adequate distance between this and any other structure.

   a. The intensity of radiant heat is proportional to the inverse of the square of the distance. In other words, if you double the distance, you would have one fourth the exposure.

   b. Not only is the control center exposed to nearby sources of fire, it can also act as an ignition source for any flammable vapors because of the ordinary electrical equipment that would be found in the control center.

   c. Adequate spacing tends to protect against fires in adjacent areas and gives clouds of flammable vapors time to dissipate before coming into contact with the immediate area of the control center.

   d. The minimum OIA guidelines are 10' between control rooms, 50' from other buildings with ordinary combustible contents, and 100' from process vessels, heaters, hot oil pumps, boilers, cooling tower, fractionating equipment, and reactors (high hazard reactors should have an added barricade to deflect a shock wave from a possible blast). There should be a spacing of 200' from loading racks, and all tanks except product storage tanks which should be 250' from the control center. Blowdown drums, flare stacks and the main gas control valve should be between 200 and 500' from the control center.

3. In the construction of a control center building:

   a. Control rooms should generally be of fire resistive construction, capable of withstanding a minimum overpressure of 3.0 psi at a distance of 100'. Monolithic walls or those having a high degree of elasticity are most desirable. Types of wall construction having this property include, reinforced concrete and structural steel. Where properly designed, reinforced concrete for walls and roof will deflect the shock wave under the influence of overpressure and resist very high loads with light to moderate damage. (Reinforced masonry, 14" brick or block walls are alternatives in descending order of desirability.) Least desirable and not recommended are the 12" brick and hollow block walls (HCB). These have little lateral resistance, and when subjected to explosive forces, can fragment to create many small projectiles which could cause further damage. These later mentioned walls will not afford 3.0 psi overpressure resistivity. The heavy construction of the room offers: Protection against shrapnel and explosion, insulation against the effects of heat, isolation against water or liquid entry into the control center.

b. The roof or floor above the computer room should be water tight slab. This slab should be sealed to the walls to prevent water from entering from the area above. Like the walls, this should be able to resist a blast or the collapse of upper stories.

c. The sub-floor space under raised floors should be adequately drained to prevent water from collecting. Water should not be allowed to collect or enter the EDP (Electronic Data Processing) room, for moisture can damage electrical wiring, instruments, and other equipment.

d. There should be at least two doors entering from the two directions of least likely hazard. These doors should be 3 hour self closing, UL-listed fire doors. All door openings into the control or computer room should be properly curbed so as to positively prevent water or any other liquids from entering the computer room through these openings.

e. Although windows are nice from an esthetic point of view to be able to look out on your process area, they are not desirable from a fire hazard viewpoint. Windows, even with the glass in them, allows radiant heat to pass easily through and into your control room. This is neither good for the men nor the equipment inside. Thermo-shock or the intense heat can shatter or melt out the panes of glass, allowing heat directly into the control room. Should there be an explosion, regular or even wired glass can be sent flying throughout the control room. If you must have windows, they should be small and of the type of glass that will pulverize rather than break into shrapnel. Under no circumstances should the windows be directly above the location where the operator would normally stand while working at either the control panel or the computer console.

f. The interior finish on the walls, drop ceilings, if any, and the raised floor should be of non-combustible construction. If wood is used it should have an Underwriter's Laboratories listed fire retardant treatment with a flame spread of less than 25.

4. Positive ventilation should be present for control rooms below the minimum spacing guidelines recommended by the OIA. This ventilation should be maintained under a positive pressure of 0.2 inches of water. Suction for this pressure should be taken by an explosion proof (Class 1, Division 1, group depending on occupancy and atmosphere) fan assembly thru a stack well above the roof. The air should come from an area which is free of potential flammable vapors.

5. By their nature, computer control equipment should be kept in a purified, cool atmosphere. The air conditioning system should take its suction above the floor, rather than near ground level if any flammable vapors in the area are denser than air. If vapors are lighter than air, suction should come from lower on the wall. A slight positive pressure is exerted by this air conditioning system. The computer area, including electric equipment and record storage facilities, should be provided with a completely separate and independently powered air conditioning system. The duct system should be independent of all other duct systems in the building. Duct systems serving other rooms of the general computer office area should have suitable fusible-link dampers at their point of entry into the computer area. Air filters of such air conditioning systems shall be of a non-combustible type. Approved products of combustion and heat detection systems should be installed in the duct systems to actuate visible as well as audible alarms and automatically shut down the air conditioning equipment in the event of the occurrence of smoke, abnormal heat, or fire.

6. Sensing elements should be provided to audibly alert operating personnel when the positive pressure falls to 50% of normal or recommended levels. Alarm and subsequent shutdowns of ventilation should occur when incoming air reaches respectively, 25% and 75% of LEL or the Low Explosion Limit. Sensing elements on LEL flammable or explosive limits should be provided within the control room and away from the make up air vents to detect and warm of the presense of flammable or explosive vapors.

7. For fire protection within a computer data/control center:

   a. A total flooding fixed HALON system, with halogenated hydrocarbon extinguishing agent, should be considered. Activation of this system should have both a manual trigger and an automatic trigger based on by-products of combustion detectors located on the ceiling and under the floor where much of the electrical conduit is located. HALON 1301, which has been classified in group 6 (the least toxic grouping and meaning that test animals can be exposed to a 20% concentration by volume for 2 hours without injury) should also be applied to these underfloor cable areas. Experimentally it has been shown that 4-8% concentration gives an adequate level for extinguishment. Total flooding Halon equipment should not be stored in areas where the ambient temperature is ever likely to exceed the range from 40F to 120F.

   Generally, these Halon systems are individually designed, and since they are a relatively new extinguishing approach, few further specific guidelines are available at this time other than the fact that the ventilation system should continue to operate for about 10 seconds after actuation of the system. The doors, windows, and ventilation system should then be kept closed until the fire area has cooled down and will not re-ignite. Manufacturer's data on Halon systems are available and should be examined. Reference to NFPA - 12A, Halogenated Extinguishing Systems is suggested. Corrosion by

Halogenated agents has been explored and found to have minimal effects. Corrosive by-products are only present when there is prolonged exposure to intense heat. Computer installations often require a tape library which contains data and program storage. Depending on the size of the tape storage area fire alarm and/or a total flooding extinguishing systems should be provided. The plastics used in tapes, reels, containers, and shields, often have flash points as low as 500F and therefore, present a hazardous fire potential. Better fire resistive plastics are being developed and should be specified for use in EDP installations whenever possible.

b. For first aid protection, it is recommended that 2-15 lb. CO_2 fire extinguishers be provided with one kept at each door. "Ordinary" dry chemical extinguishers are not recommended because D.C. fire extinguishers are not meant to handle Class A deep seated fires such as paper or wood. An ABC powder which can handle all but metal fires leaves a sticky residule which would have to be cleaned from each electrical contact within a computer or control panel.

8. For the electrical system within the control room, the power supply for the computer equipment should be completely separate from the air-conditioning system and should be de-energized by a separate emergency "power-off" control or master shutdown switch. Such push-button controls should be placed in a convenient location preferably near the operating console and/or next to the main exit doors. It is recommended that auxiliary emergency controls(glass enclosed) be provided in duplicate outside the air conditioned computer room, to permit shutdown or either the ventilator units or the computer systems from a remote point. Protection against lightning and line surges should be provided as well as battery operated emergency lighting units.

Wiring throughout the computer room, including that beneath the floor, should be in accordance with the National Electrical Code. Power and signal cables should be fitted with water tight receptacles and should be well separated for ease of access and replacement. No special fire protection is required where such cables are separated by non-combustible barriers or metal raceways. All wiring and component plastic parts comprising the construction and assembly of the various units of the computer equipment and data processing system should be of a thermally stable composition to meet the normal operating temperatures of the various units and be flame retardant. Wire and cable insulation should be self-extinguishing, especially when massed wiring configurations can generate enough heat to cause ignition and propagate combustion.

9. Housekeeping within the control room:

a. Combustibles such as rags, charts, articles of clothing, boxes, storage of samples, etc. should be kept away from control panels and consoles.

b. Consideration should be given to storing vital stocks of standby records and master data media (including plastic or metal base electronic and electrostatic tapes, memory drums, memory cores, etc.) in a separate room employed only for this purpose and suitably guarded with automatic fire protection. Water tight, fire resistant, heat insulated, non-combustible containers, and cabinets should be considered. Where sprinklers are used, they should be equipped with water-flow alarms to a continuously attended location within the plant.

c. Current records to be handled in the computer room should be kept to a minimum and in quantity to meet only the daily operating needs. Commonly encountered paper records, written programs, punch cards, carbons, spent forms and other unwanted stationery and other waste combustible material should not be permitted to accumulate in the computer and record storage rooms. Proper facilities for their storage elsewhere or their disposal should be provided on a daily basis.

B. Instruments in process areas are the eyes and ears of the plant. Safe and accurate operation of modern refinery units depends, in a large measure, upon proper instrumentation. Each process should be analyzed for suitable instruments, alarms, and controls for emergency conditions, as well as for startup, shutdown, and normal operation. Equipment for automatic startup or shutdown sequences should be carefully reviewed.

1. Of particular importance is the effect of power failure. Auxiliary automatic equipment should be provided to enable an orderly shutdown (if necessary) in case of the loss of power. This would include standby or auxiliary supplies of essential utilities, such as electrical and instrument air supplies. Controls should be provided to minimize shutdowns on momentary power interruptions.

2. All instruments should fail safe. That is, instrument failure should cause controlled equipment to automatically remain in position, open, close, start, stop, or do whatever has been predetermined as necessary to continue safe unit operation. Particular care must be taken to insure that should any group of instruments fail, they will as a group fail safe! It is possible for instruments to individually fail safe while as a group fail in an unstable and dangerous manner.

3. Avoid the use of instruments in dual or multiple service if operator confusion can cause unsafe conditions. In any case, separate indicators must be used for each specific alarm point.

4. Process control instruments of critical loops should be arranged so that a specific deviation from set point, will activate a visual alarm (preferably a flashing light). Further deviation will result in an audible alarm. Still further deviation from this point should result in the actuation of an automatic shutdown procedure.

5. Visual sequence annunciators or print-out devices should be employed when it is necessary to determine the proper sequence of failures of related equipment.

6. Instruments must be made of materials suitable for the service, particularly when subjected to corrosive, erosive, or high temperature conditions.

7. Generally, instruments should be located so that they can be operated and serviced from grade or a convenient platform--not from ladders or scaffolds.

8. Instruments should be calibrated or checked at regular intervals. Secondary cross-checking instruments should be available for use. Regular intervals or instrument checks may coincide with the process unit turnaround.

9. Alarms and shutdowns should be capable of being checked while on stream without the actual upsetting or the shutting down of a process.

10. Hydrocarbons or other flammable toxic fluids or vapors should not be piped into control rooms for instrumentation. In general, pneumatic or electrical signals should be used. There should be no common flammable vapor and pneumatic control lines. Check valves are an insufficient safe guard to prevent a back up of flammable vapors into the pneumatic control lines. With a blast resistant control house, should the flammable vapors vent into the structure, the entire building could be just one large bomb. Tubing bundles, instrument ducts, and conduit must be equipped with vapor seals and vents to prevent process area vapors from entering control rooms and instrument cases. Data links or instrument cables should be suitably protected from fire exposure by running these leads in fire resistive cable trays.

11. Pneumatic instrumentation should be provided with an auxiliary air supply in the event of a fire or another emergency situation that destroys the primary source of air.

12. Outside process instruments should not be enclosed in combustible instrument housing.

13. Panel boards in control rooms should generally use high density instruments in order that space may be conserved on the panel. Panel boards should be designed to display all the pertinent information necessary to control and monitor the process. Instruments, alarms, flow diagrams, etc. should be well laid out in order that a process may be easily followed. Control loop indicators should be logically arranged to best accomplish this objective.

C. Computer Control in process areas is the heart.

1. First direct digital control:

Because the digital computer has direct control over process control loops with no operator action necessary, the first recommendation for a proposed digital machine would be to obtain a system which will afford a great deal of reliability. Properly designed systems will achieve adequate process control and/or optimization with a small amount of operator supervision.

a. Primary consideration should be given to the provision for digital computer backup (i.e., a spare computer which could back up the main on-line computer). The need for such a device becomes more important when one considers the fact that loss of the master may cause the process to continue uncontrolled (unless hardware and manual backup is provided). The spare "slave" is further justified by the fact that it can perform business, scientific, or report generating operations when it is not on-line to the process. If such a slave is provided, it is an extremely important recommendation that the master, via data links, continuously update the memory in the slave for set point changes, abnormal process deviations, and other pertinent information so that transfer of control from master to slave is nearly instantaneous.

b. Manual and hardware backup devices should be provided on critical process loops in a DDC installation. For critical loops, an "inline" system can be set up, showing the value of the measured variable, with a manual, "raise-lower" valve control on the same display. This in-line arrangement would not operate in this case until called on to do so, as in an emergency situation. Deviation type indicators can be used on the measured variable display, whereby value signals are set up on potentiometers and backed off against the measured variable signals. These critical hardware backup devices should be continuously brought up to date automatically by the master computer.

c. All electronic cabling (analogous to pneumatic cabling for analog controllers in conventional control and supervisory computer control) should be installed not only in fireproofed cable trays which protect against excessive heat, moisture, and mechanical damage, but also in such a manner as to avoid coupling with sources of high intensity electrical transients. The noise or interference signals which can be picked up from these sources can cause erratic process control. In order to reduce the high intensity electrical static, intercabling practices (putting cables of similar current and field generation together) should be grouped in the following categories.

1. power wiring
2. control and intersystem wiring
3. digital inputs
4. analog data wiring
5. digital outputs

Major wiring diagrams should be reviewed prior to installation. Grouping the cables in like categories will tend to avoid interference problems. Reduction of this interference will result in more efficient and safer process control.

d. Grounding of individual electrical equipment is often performed. However, in DDC it is necessary to ground all equipment at one point. Multiple grounds will introduce undesirable ground loops because the individual ground loops are not at the same potential. These multiple grounds will cause incorrect data input signals and return outputs. For effective and safe control, these incorrect signals should be eliminated. Grounding of signal and power leads, electronic shielding, and miscellaneous electrical equipment grounding procedures should be developed with the Insured prior to installation. Specifically, signal and power leads should be grounded at the source only, shields must be grounded close to the source, and if electrical equipment is grounded to main site ground or the "tree" system, ground wires should be at least No. AWG 0000, or a copper bus with a 1 square inch cross sectional area.

e. While DDC could control a process as programmed without regular operator intervention, communication devices (typewriters, CRT display, analog display, tape, etc.) should be provided to inform the operator at periodic intervals as to the status of the overall process. This periodic listing should be a summary report of the previous operating period and should include any unusual process fluctuations which may be the initiating of trends. This would enable an operator to detect such trends and make preparation for corrective action prior to any upset.

f. It is imperative that a preventative maintenance program be initiated to periodically inspect the entire DDC system insuring that equipment failures are kept to a minimum. The current reliability of DDC systems demands that such a program be followed.

g. An auxiliary power supply should definitely be provided in event of power failure. Generators should be capable of producing sufficient electrical power to run the DDC system long enough to effect total orderly plant shutdown. Battery banks could also be considered as a source of much needed electrical power. Emergency power should also be provided for control center air conditioning, fire protection system, etc. Battery powered emergency lights should also be installed. One possibility, a separate gas turbine generator, for instance, could be considered for the installation.

2. Second supervisor control:

As stated previously, supervisory control utilizes conventional analog controllers so there is no need for hardware backup devices or a spare computer to go on stream in event that the master supervisory unit fails.

a. In event of any computer failure, control instruments should be selectively wired so that they will take over instantaneously from the computer. This should apply to closed or open loop computer control. In event of computer programming error, where control is still in effect (though incorrect process changes are made) the operator should have the option of taking over control from the computer.

b. The preventative maintenance program described above for DDC is likewise applicable here, elaborated as follows:

aa. Since computer or operator control of plant process' can only be effective when correct readings are produced by sensing, measuring or recording instruments and there is the correct response by controlling equipment, it is essential that a detailed, systematic program of testing and maintenance be followed for all these devices.

bb. Computer cabinet and circuit layout design should permit rapid and convenient trouble-shooting and maintenance in event of computer failure. Maintenance of the computer is considerably more important than any additional expense incurred to provide adequate space requirements needed for convenient maintenance.

cc. In process' that are particularly hazardous and could be a threat to the safety of the plant and its personnel, the sensing and measuring devices used in process control should be duplicated so that any partial malfunction of one will not permit a dangerous condition to go undetected. Valves or other control devices should be arranged to fail in a safe position. All instrumentation should be located so that inspection and maintenance of the devices are safely and readily ahcieved.

dd. Since most leased electronic computers require a period of scheduled preventive maintenance, equipment of this type which is purchased outright should also receive the same type of preventive maintenance either by the manufacturer or personnel trained by the manufacturer.

IV. In Conclusion:

The central control room is an integral part of all modern plants. The grouping of recording and controlling instruments facilitates the work of operator and concentrates responsibility for the operation of the plant. Although the first cost is high, it may reduce the number of operators required to man the plant and provides an appreciable saving in operating cost. In dealing with emergency shutdowns, such as those arising from explosions and fires, the facility afforded by centralized control is vital. Centralized control is especially important for controlling a specific unit or chains of units. A central control center employing conventional controllers is not recommended for control of an entire large non-integrated plant for the obvious reason that if a fire or explosion in the one control room occurs, it will adversely affect process control in the entire plant. We recommend for conventional process control houses, that the OIA spacing guidelines be strictly followed.

Plants on supervisory (with or without optimization control) or direct digital control should have data centers well spaced from the nearest process unit. 100 feet is the minimum allowable spacing for such a data center. Computers should <u>not</u> be installed in conventional control houses because of differences in construction of the structures.

APPLICATION AND FUNCTIONAL TEST OF SELF-CHECKING PROGRAMS; THEIR INFLUENCE ON THE FAILURE PROBABILITY OF COMPUTERIZED SAFETY SYSTEMS

by

H. Schüller
Laboratorium fur Reaktorregelung und Anlagensicherung
Technische Universitat Munchen

ABSTRACT

Computer-self-monitoring programs turn out to be appropriate to ensure the indispensable reliability of computerized safety systems. A practical test of a realized program system showed that every dangerous component fault can be detected and made fail-safe. Thus the reliability as well as the availability of a computerized system may increase about several magnitudes. The influence of the tested fault detection time on the failure probability of a 2-out-of-3 reactor protection system in case of a shut down is shown and discussed. Furthermore, the limitation of the achievable reliability by the completeness of the test programs - i.e. the portion of the recognized from all possible faults - is explained in detail.

1. INTRODUCTION

The possible failure effects caused by a component fault in the central unit of a process computer depends in most cases not only on the type of the fault but also on its entry moment. According to the momentary program state, the component fault may have very different effects on the computer controlled process. There may occur dangerous or harmless effects or no effects at all. Without taking special precautions it is therefore normally impossible to predict the special accompanied faulty action. This means that we must assume, for the present, the possibility of dangerous effects from each component fault, if we delegate such important tasks as reactor protection to a process computer /1/.

The concerted acting of two measures, however, enables us to detect every dangerous component fault and make it fail-safe:

1st the dynamic lay out of the computer outputs by using supervision units /2/,

2nd using computer self-checking programs /3/.

The pulse supervision units ensure that their safety technical measures will be initiated as soon as the supervised pulses change their frequency more than an allowed bandwidth. The task of the computer self-checking programs is to cause such a frequency change, if a computer fault had occured, for example by cutting off all further output pulses /3/.

2. USED TEST PROGRAMS

In /3/ some methods and test programs for computer self-checking have been suggested to detect failures of the central unit within a short time. The realization of these ideas led to a test program system consisting of the following single programs:

1st Special function test programs
   a) instruction test program
   b) core store function test program
   c) input/output function test program

2nd Global supervision programs
   a) core store constant data test routine
   b) program flow monitoring routine

These programs are running partly in a fixed cycle in the foreground (instruction test; i/o-function test part 1: electronic; program flow monitoring) and partly in the background (i/o-function test part 2: relays; core store tests) within the free cpu-time. It is planned to use this test program system at the reactor safety and the control rod computers of the nuclear power reactors at Brunsbuttel and Phillipsburg. So it will be guaranteed for these computers that component faults are detected either within 200 msec by the foreground supervision programs or within 60 sec by the background supervision programs.

3. THE INFLUENCE ON THE FAILURE PROBABILITY OF A COMPUTERIZED SAFETY SYSTEM

Let us first have a look at the failure probability of a reactor protection system (2-out-of-3 valuation logic), when we suppose that a (dangerous) failure cannot be detected until the next scram occurs. The mean time from failure occurrance until its detection is then half the mean time between scrams (e.g. 1/2 . 100 days). The corresponding failure probability is shown by the curve in fig. 1 /4/ depending on the mean time between failures (MTBF) of a single computer. We see that our system is too unreliable, if we do not take special precautions for rapid failure detection.

If we ensure, however, on the one hand that component faults of the single computers are recognized as quickly as possible and on the other hand that immediately after failure detection it is switched over to a safe situation, we are able to reduce the failure probability of our system considerably /5/. Fig 2 shows the achievable failure probability for various failure detection times. For comparison, the curve of fig. 1 (no special failure detection) is drawn in once more. We can see that the above mentioned failure detection time of 200 msec and of 60 sec respectively are in all cases sufficient to ensure satisfactory reliability.

The curves for the non-availability of the system are identical with those shown for the failure probabilities if we replace the parameter "Failure Detection Time" by the parameter "Sum of Failure Detection Time and Repair Time". In this case failure detection times become relatively small, leaving only the repair time responsible for the non-availability. Our failure detection times therefore increase the availability up to the limitation set by the time to repair.

4. THE INFLUENCE OF THE COMPLETENESS OF THE TEST PROGRAMS ON THE MEAN FAILURE DETECTION TIME

It seems to be hardly possible to show that the computer self-checking programs are really complete, i.e. detect every possible dangerous failure. As the very long detection time before the next scram sets standards for the non-detected portions of all dangerous failures, this portion may limit the achievable system failure probability.

Corresponding to this, a mean failure detection time can formally be calculated if different detection times exist for various failure modes and if the portions of these modes of all possible failures are well known. Fig. 3 shows the curve of this mean failure detection time as function of the portion of the non-detected failures (failure detection time = 1/2 scram interval). For the detected component faults a detection time should exist which is the parameter in fig. 3 for the different curves.

We can see, that the mean failure detection time is limited obviously by the actual failure detection time of the self-checking programs. This applies even when the latter are absolutely complete and detect every failure. On the other hand the portion of the non-detected failures limits our mean failure detection time also, even when all detected component faults are recognized very quickly.

From this we can draw the following two conclusions for the practical function test of failure detection routines:

1st It does not seem expedient to show by tests a better

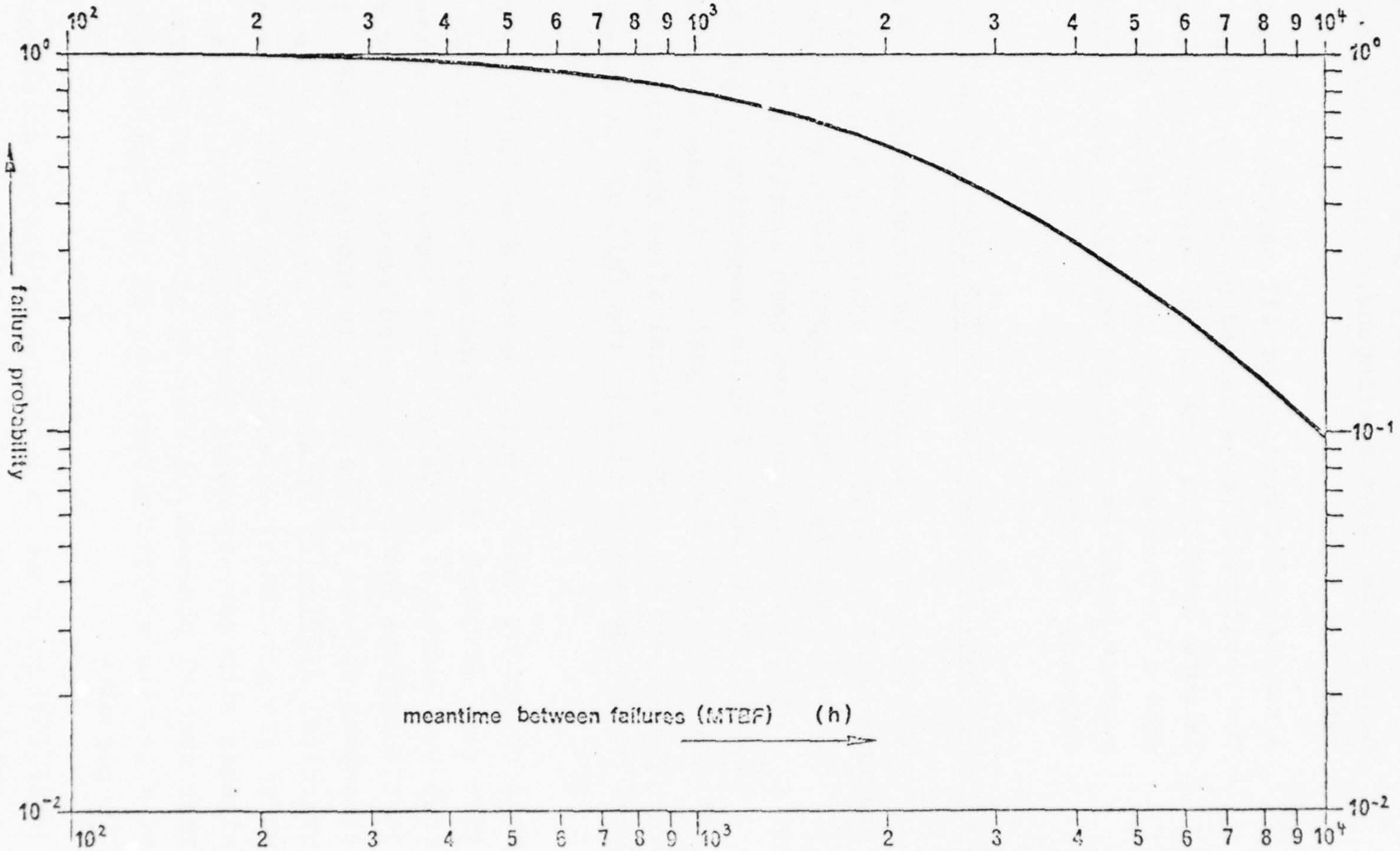


FIG.1 Failure probability in case of a shutdown demand for a computerized 2-out-of-3 protection system dependent upon the MTBF of a single computer, if no special precautions for rapid failure detection are taken.

portion of detected failures than is indicated by the sharp bend of the curve corresponding to the achieved failure detection time.

2nd A non-detected portion of all hardware failures limits the positive effects caused by the shortening of the failure detection time on the system failure probability. Thus a further shortening of the failure detection time becomes pointless without enlarging the portion of the detected failures.

5. PRACTICAL FUNCTION TEST OF THE SELF-CHECKING PROGRAMS

It had been intended to prove the functioning of the developed computer self-checking program system by a practical test. To do this all possible static signal faults of the central unit of the AEG 60-10 computer have been simulated by forcing statically 0 Volt and + 5 Volts respectively upon each single integrated circuit output signal. This was done during the program run on one output signal after the other, measuring the failure detection time by the fall of the pulse supervision relay.

This component fault simulation could be done in a non-destructive way, although the TTL-Chips operate above capacity when they are forced at +5 Volts. The computer cards used for this test therefore should not be used again in reactor safety computers because of the possibly shortened MTBF. The next practical difficulty arises from the fact that the circuits which are not really defective try to switch their output signals with corresponding changes in input levels. This means that special precautions have to be taken to ensure the forcing of a certain static voltage even at the high frequencies of the TTL-circuits.

Other failure modes, e.g. wiring or layout disconnections or signal bypass, cannot be simulated in a non destructive experimental way.

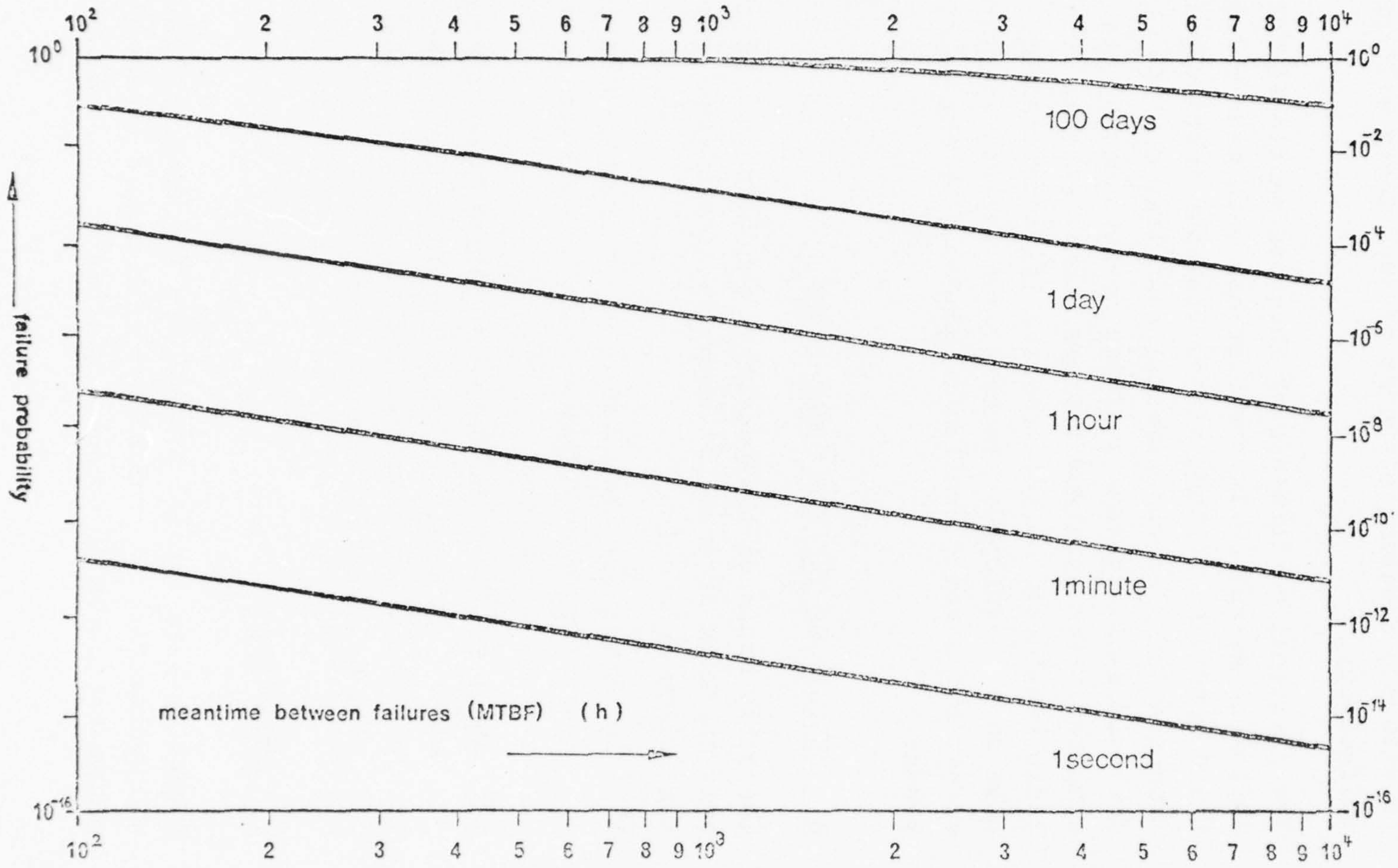


FIG. 2 Failure probability in case of a shutdown demand for a computerized 2-out-of-3 protection system dependent upon the MTBF of a single computer, if a limited detection time for all component faults exists which is the parameter in this fig.

If there exists a limited system failure probability which can be allowed, a definite subset of such tests will prove adequate reliability.

The practical performance of theabove mentioned failure simulation at an AEG 60-10 computer has been made by the Kraftwerk Union according to suggestions from our institute and demands from the TUV. An experimental test like this is only practicable for such small systems as those represented by the KWU-safety computer. For enlarged systems other methods, such as a complete simulation of the computer logic on another computer have to be developed. In this case you can simulate any failure modes, but a great problem is produced by the calculation time needed.

6. RESULTS OF THE PRACTICAL TESTS

The performance of the practical function test of the computer self-checking programs by component fault simulation furnished the following results:

1st The final version of the test program system detects every simulated component fault which can have undesired effects on the program run.

2nd About 97.5% of the detected faults have been recognized within 200 msec, the rest within 60 sec. The mean failure detection time for the detected component faults therefore will be about 850 msec.

7. CONCLUSIONS

Without taking special precautions for fast failure detection, the system failure probability with regard to possible dangerous component faults of computer systems is too small.

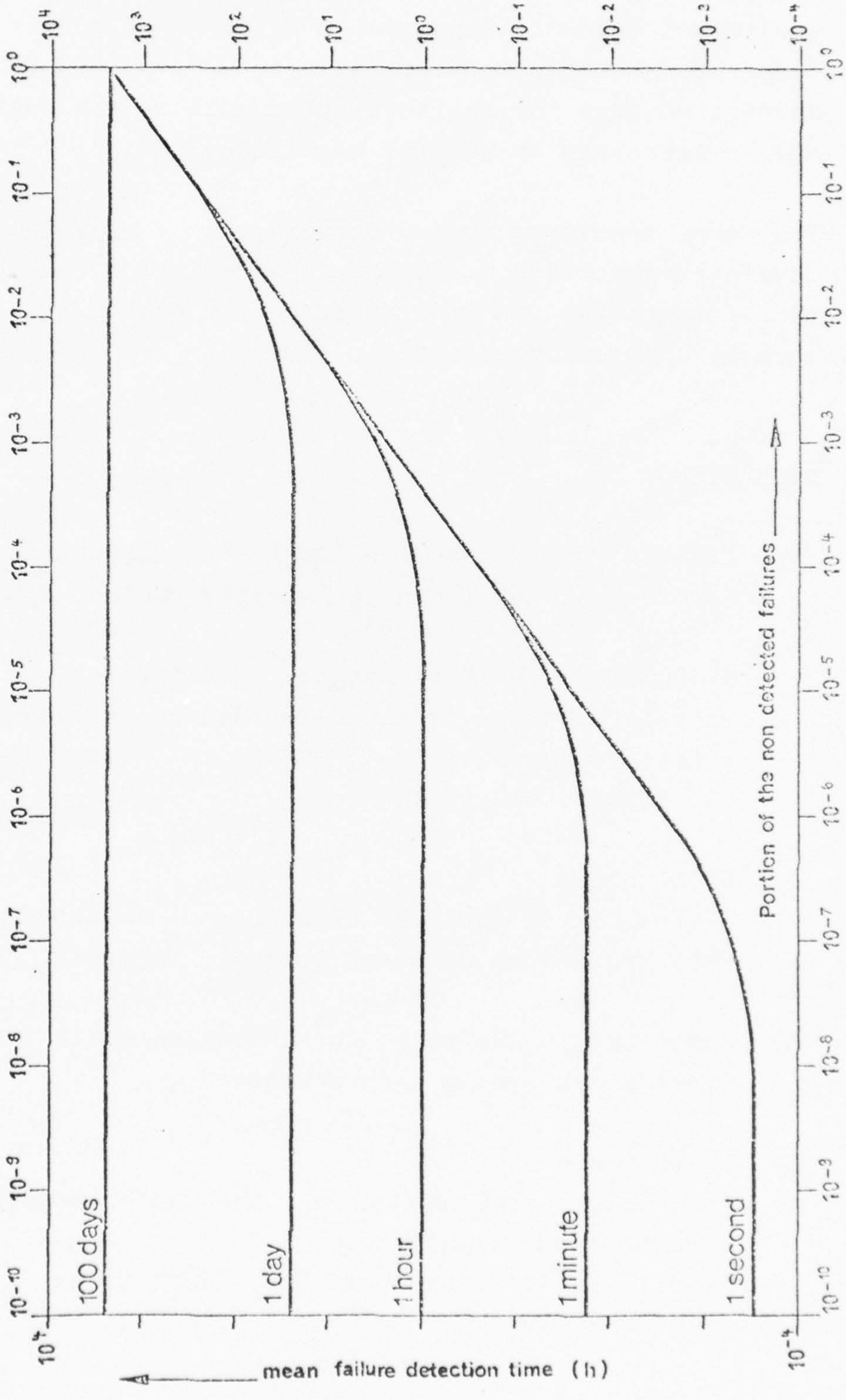


FIG.3 Calculated mean failure detection time dependent upon the portion of the non detected failures. For the detected component faults a detection time should exist which is parameter in this fig. (Meantime between scrams = 100 days).

Computer self-monitoring programs - with the aid of simple additional hardware supervision units - may detect every dangerous static component fault in a computer within such a short time that the reliability as well as the availability may be increased by several magnitudes.

The above mentioned test program system - especially if developed further - may be of great help for fault localization (diagnosis) and thus increase the system availability even more by shortening the repair time.

LITERATURE

/1/ Einsatz von Prozessrechnern in Reaktorschutzsystemen.
Report MRR 124, Techn. University Munich, April 1973

/2/ H. Schuller, W. Ehrenberger, H. Biegel
Taktuberwachungseinheiten fur den Reaktorschutz mit Prozessrechnern.
Elektronik, 11, 1973.

/3/ H. Schuller
Self-checking features of a process computer.
EHPG Meeting on Computer Control, Volume II, April 1973.

/4/ Kommentar zum Beitrag "Zur Zuverlassigkeit von Prozessrechnern im Reaktorschutz"

/5/ H. Hoermann
Reliability problems through the use of computers in reactor protection systems.
Proc. IAEA Symp. on Nucl. Pow. Plant Contr. a. Instr., SM-168/D3, 1973

EUROPEAN PURDUE WORKSHOP
Safety and Security

Author: 1. H. Schuller
2. W. Schwier

Institution:
1. Laboratorium fur Reaktorregelung und Anlagensicherung, Garching
2. Bundesbahn-Zentralamt Munchen

Date (assigned): 16 Dezember 1974

Date (completed):

TC "SS"	
Nr:	6
Category:	T
Updates:	None
Replaces:	None
pp:	7

Title: Safe Computer Systems Hardware - Part 1

1. Aim

Many technical processes, if they are controlled incorrectly or if they are not sufficiently supervised, can dangerous for man, equipment or products: Trains of a railway could collide or derail if controlled wrongly; nuclear power plant could be be perilous for human bei if it is not switched off before critical situations occur; a chemical plant or a rolling-mill could be destroyed if incorrect control commands are given. A computer system controlling or supervising a dangerous technical process must not give control commands which are adulterated in a danger producing way. Even the l of control commands could, under certain circumstances lead to danger.

A computer system may be considered as working satisfactory

1. if the tasks it has to fulfill are defined in a correct and complete manner,
2. if these definitions are transformed into the logical concept of the equipment (hardware and software) in a correct manner,
3. if all components are dimensionally correct and if environmental influences are taken into account,
4. if there are no manufacturing defects,
5. if the equipment is installed correctly at the site,
6. as long as no component fails and as long as disturbing environmental influences do not exceed the permissible value,
7. if no mistakes are made during maintenance

In the following we shall examine how hazardous occurences mentioned under 6. can be avoided. We assume that the requirements 1. to 5. and 7. are satisfied. Therefore, the fundamental problem is: How can incorrect control signals be avoided, even if components in the computer fail or diturbing influences become effective.

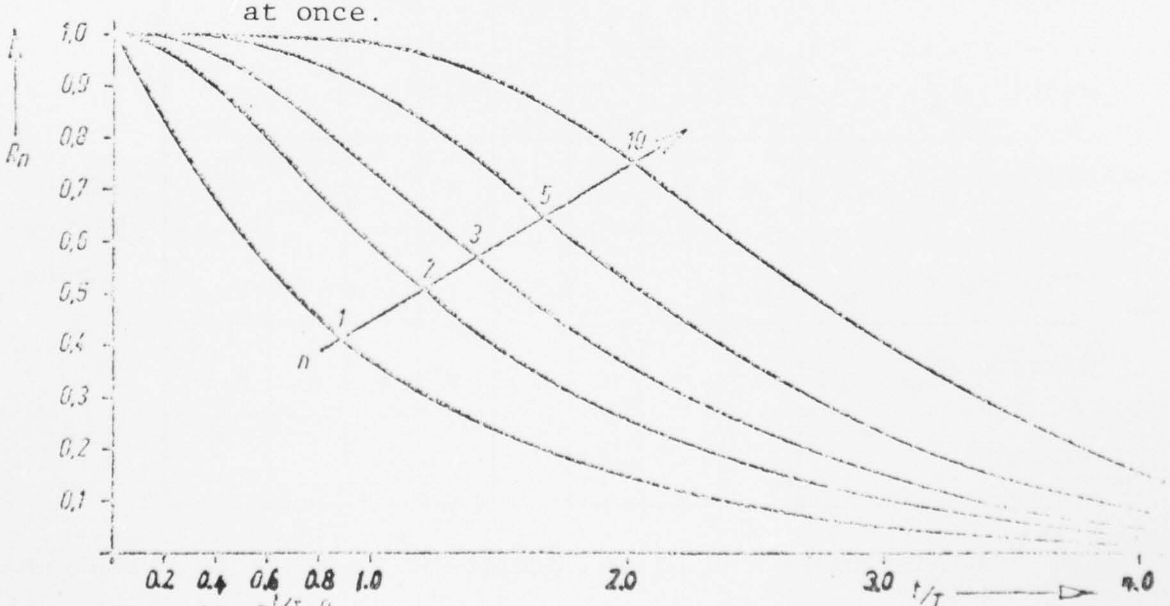
2. Method

There are two methods with the following differentiations:

1. There must be a very high degree of probability that the computer system will not fail for a certain period (operational period).
2. There must be a very high degree of probability that component failure and disturbing influences fail on the safe side.

2.1 Safety through reliability

The first method is to be used if the controlled process does not have a safe side, as is the case in aviation and astronautics. High reliability can be achieved by selecting very reliable components and by installing redundant spare units. Fig. 1 shows the probability of survival in the case of a degree of redundance of 1 to 10. T.is the mean time between failures (MTBF) of the individual non-redundant unit. We see that only during short periods the probability of survival is sufficiently close to 1. Therefore, before the start of an operational period, it has to be assured by means of a comprehensive test, that the units work correctly and that the probability of survival still has the value 1. This approach is sometimes termed the check-out philosophy. At the end of an operational period the probability of survival can be lowered by only a tolerable small value, in order that dangerous occurences during the operational period will be almost impossible. This limits the duration of the operational period. However, if a continuous operation is required, each one of the redundant units must be checked regularly with regard to failures; and when detected it has to be repaired at once.



$R_n = 1 - (1 - e^{-t/T})^n$

n = Factor of redundancy

T = MTBF

Figure 1

PROBABILITY OF SURVIVAL FOR A REDUNDANT SYSTEM WITHOUT REPAIR

In the following we shall particularly examine random failures and their effects. Let us start from the assumption that systems are free from faults when put into operation. The faults in the computer system may

- occur in a static or transiet manner
- be single or multiple
- be dangerous or not dangerous
- be obvious or remain unnoticed

	dangerous Combination							
Failure	1	2	3	4	5	6	7	8
static	x		x		x		x	
transient		x		x		x		x
systematic	x	x			x	x		
stochastic			x	x			x	x
single	x	x	x	x				
multiple					x	x	x	x
dangerous	x	x	x	x	x	x	x	x
not dangerous								
unnoticed*	x	x	x	x	x	x	x	x
obvious*								

Table 1 Failure modes

*before using the failed component in a safety operation

Faults occuring in a transient manner are in general the result of faulty design (e.g. crosstalk of signals in the computer in the case of particular patterns) or the result of environmental influences (temperature variation, vibrations, stray effects, corrosion etc.). They can, therefore, be reproduced if the adequate limiting conditions are observed. Multiple faults have to be taken into consideration, expecially if they have a possible common cause (consequential faults, common-mode faults). But also random multiple faults can not be excluded a priori. Like single faults they have to be made inoffensive; unless an adequate function of the circuit-network has made their occurance so unlikely that they can be neglected. Table 1 shows all the dangerous combinations of failure modes. In the columns the types of failures are to be seen. For instance the column 7 is the static multiple failure, which becomes dangerous. It remains unnoticed before the safety operation occurs and the cause of this failure is stochastic. A fail-safe system had to be constructed in a manner, that none of the 8 types of failure has a dangerous effect on those signals leaving the system.

4. Possibilities for fail-safe computer systems

4.1. Fail-safe separate computers

Basically, it would be possible to construct a special computer consisting of fail-safe circuits. Furthermore, the use of normal components is being taken into consideration, but in this case the storage, transport and processing of information would be in coded form. Processing and transferring the information in a coded form would demand a special computer, built for this purpose. The fault detecting characteristics of the codes have to recognize component failures or active disturbing influences, so that the faulty elements could be switched off.

Presently we know of no solutions for a fail-safe computer for industrial applications. We shall therefore examine the problem how to achieve a fail-safe operation with computers which are not fail-safe. For this purpose we shall begin with standard computers used for industrial applications.

Proposal for the next chapters

4.2. One none fail-safe computer, operating in a fail-safe manner - description.

4.3. Fail-safe computer systems, operating in a valuation logic - description.

4.4. Comparative discussion of 4.2, 4.3.

5. Details of 4.3
   - The importance of single and multiple failures
   - Methods to achieve a very small probability for random multiple failures
   - Self-checking programs
   - Systematic multiple failures
   - Should be assumed, all multiple failures being dangerous?

Remarks to Revision of "METHODS TO DEVELOP SAFE COMPUTER SYSTEMS"

by

H. Trauboth

1. Approach for Revision of Paper

1.1 Rationale

- Concentrate in a systematical way on safety aspects in all phases of development process as outlined in version 1.
- Include "safety methods" in the design of computer systems.
- List the important safety measures but leave an evaluation of these measures with regard to different safety requirements to future investigations.
- Propose project management measures for safety-conscious development.
- As a prerequisite, the development of a safety-oriented system must follow sound design and project management rules.

1.2 Definition of Safety

- Safety measures should ensure that any error in computer hardware and/or software does not cause harmful or unpredictable actions.
- It is assumed that errors can occur at all phases of the development process. (Fig. 1).
- At each phase, the following should be checked:
  - a) error prevention (protection)
  - b) error detection
  - c) criticality of error
  - d) actions or consequences in case of a harmful error (error recovery actions)

- Criticality requires determination of consequences of an error (grades of criticality).
- Types of errors:

o Requirements errors	RE
o Hardware errors	HE
o Design errors	HDE
o Implementation errors (physical wear)	HOE
o Software errors	SE
o Design errors	SDE
o Implementation errors (Program errors-bugs)	SPE
o Documentation errors	DE
o Communication errors	DCE
o Printing errors	DPE
o Interface error (between hardware and software)	IE

- Some checks may be common to all phases, others are unique to each phase.
- Types of tests during each phase of the development process:
  - a) We test at each phase if proper means for error prevention, error detection, determination of criticality of error and error recovery actions have been built into the design to be executed (exercised) during the operational phase.
  - b) We test at each phase if errors in the design of that phase have occurred. We determine the criticality of these errors and take actions to eliminate these errors. (During "Reviews" under Project management.)

1.3 Remarks

- Version 1 together with comments by Dr. Ehrenberger and Mr. Taylor are used as a basis for version 2.

- o Expand version 1 where necessary (e.g. project organization) and reduce it where feasible.

1.4 Summary of Comments

a) Comments by Dr. Ehrenberger refer to:

- o Error evaluation
- o Costing of design and safety measures
- o Environmental effects on design
- o Design evaluation (bottlenecks in data transfers
- o Change control
- o Hardware testing
- o Project organization (Test team)
- o Maintenance

b) Comments by Mr. Taylor refer to:

- o Design philosophy of safe systems
- o Structuring the design
- o Structure of project team
- o Costing of safety measures

c) Comments during discussion refer to:

- o Prevention of human errors caused by operations personnel by organizational and procedural means.

2. Examples of Approach - see "Software Detail Design" (of version 1)

a. Error Prevention (SDE)

1. 2. level of structured programming
   - o standardized interfaces for program control (e.g. parameter transfer between subroutines)
   - o access to data files via file access handler
2. Unique but descriptive naming of labels, addresses, variables and data fields

3. Descriptive commentary to program statements (→documentation)
4. Use of reference indices in documentation to obtain consistency between various levels of design (→documentation)
5. Lowest level module size not more than 50 statements
6. Nesting of loops via explicit stacks.

b. Error Detection (SPE, HOE)

1. Check numbers for each program step (relay runner) of program logic
2. Check code for data access.
3. Provide range, limit and plausibility checks
4. Check for critical timing requirements of execution of a subroutine, data transfer and data transmission.
5. Comparison of two different arithmetic programs for the same algorithm
6. Redundant program operations and comparison.
7. Redundant storage of data and comparison

c. Determination of Criticality of Errors (SPE, HOE)

For each of the error detection measures bl...bk determine the consequences and their criticality, if no recovery action would be taken, e.g.

- no harmful action
  - e.g. no protocol of unimportant data is printed or a less important subprogram is bypassed.
- harmful action (low criticality)
  - e.g. monitoring of important temperature guages does not take place, however, it is knows by law of physics that temperature cannot change rapidly (long range effect)
- harmful action (high criticality)
  - e.g. a control rod will be activated erroneously

d. Error Recovery Actions (on detail level)

For each of the error detection measures bl...bk, one or more unique or common error recovery actions are possible, e.g.

- o repetitions of erroneous operation (e.g. in transmission or arithmetic error in case of sporadic error)
- o in redundant operations:
    - o switching off the operation which was determined faulty by majority voting and continued operation with reduced redundancy.
    - o stopping all redundant operations and
        - o continuation of operation with reduced capacity (graceful degradation).
        - o stopping completely all operations
- o initiation of alarm message and waiting for operator action based on options that are printed out
- o switching in back-up device

See "Functional Systems Design" (of version 1)

1. Control Strategy (Main Control)

a) *Error Prevention*

- o Fixed time slots and length of time allocated to each task (fast cycles, slow cycles), i.e. pulsed synchronous operations if possible (HE,SE)
- o Task initiation by polling rather than by interrupt (also for asynchronous operation) (HE,SE)
- o For asynchronous operation, provide handshaking control (HE, SE)
- o Avoid long transmission lines with high data rates. Perform as much prepocessing at data source as possible (HE)
- o Define functions and subfunctions in such a way that
    - o each function has its own files

  - data traffic between functions is a minimum (weak coupling between functions)
  - major functions are on separate hardware devices, e.g. data acquisition and preprocessing (limit checking, plausibility checking)
- Separate clearly between data flow and control flow
- Use pulsed hardware units
- In decentralized systems, give each major subsystem the capability to take over a minimum of overall control in case of failure in controlling subsystem. Assign one subsystem as the main controller.
- Provide separate hardware lines and hardware check units for checking basic functions of peripheral devices, e.g. power supply.
- Use hardware "coordinator" for synchronization of data transfer between processors in multicomputer systems (see Wobig).

b) Error Detection

- Provide checks for proper time allocation and length of operation of tasks in design (HE,SE)
- Provide pulse-rate detectors.
- Provide special software functions on
  - hardware error detection (test programs),
  - error recovery,
  - software error messages and protocol.
- Provide redundant units and voters.
- Provide back-up units, files and programs.

c) Determinationsof Criticality of Errors

In Relation to b. determine consequences of errors, e.g.

- for each task determine criticality of time loss, e.g. fast changing pressure measurement in dangerous pipe must be processed in time while slow changing environmental temperature may be shipped from time to time.

d) Error Recovery Actions (on functional level)
see d) of "Software Detail Design"

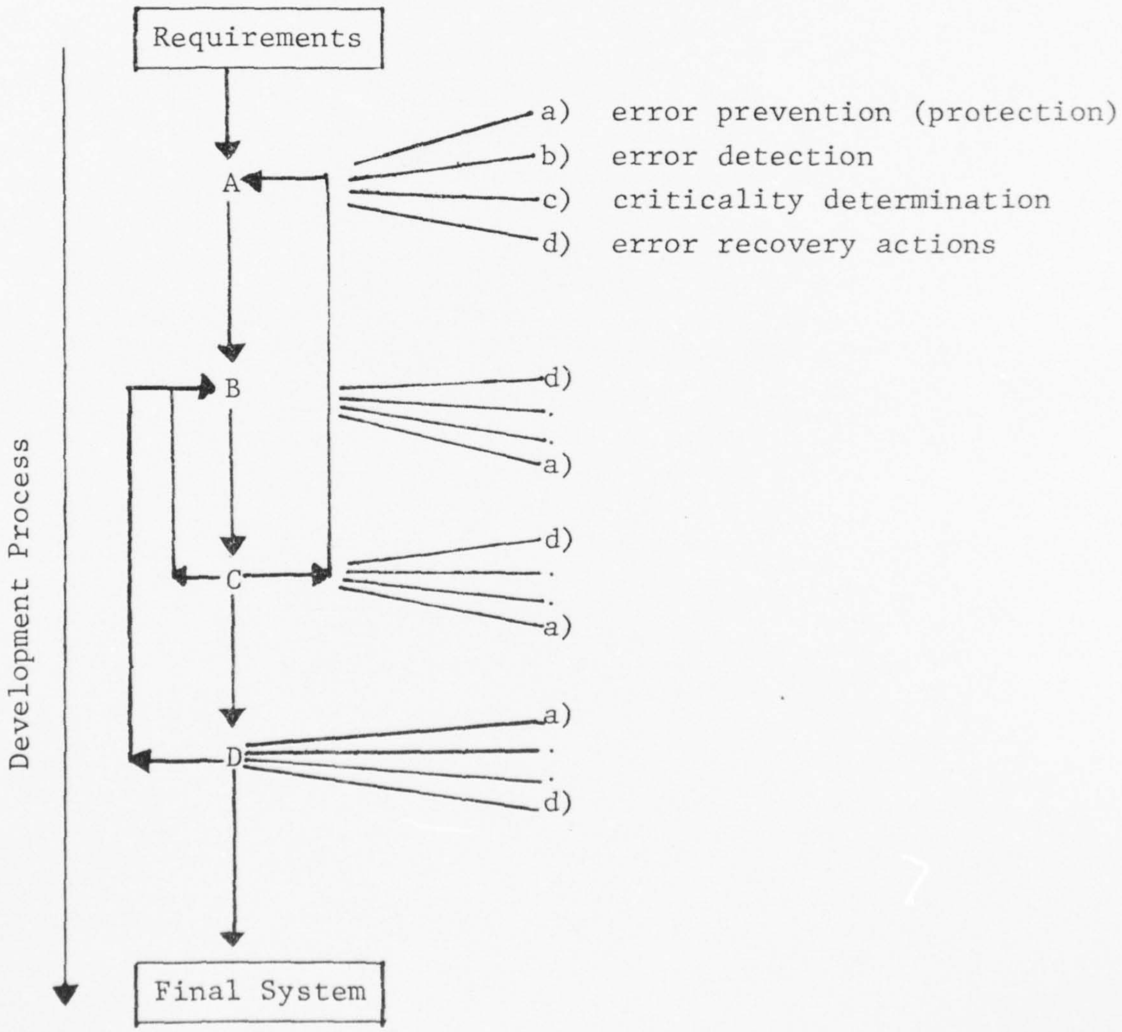

Requirements
a) error prevention (protection)
b) error detection
c) criticality determination
d) error recovery actions
A
B
C
D
d)
.
.
a)
d)
.
.
a)
a)
.
.
d)
Development Process
Final System

FIGURE 1

ERROR CHECKS DURING DEVELOPMENT PROCESS OF COMPUTER SYSTEM

COMPUTER SAFETY AND SECURITY

BACK TO BASICS

by

J. R. Ellison

The National Computing Centre
Manchester
U.K.

Presented to:

The European Purdue Workshop
T.C. on Safety and Security
in June 1975

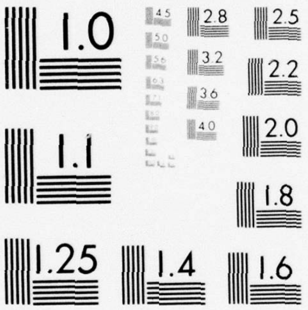
AD-A036 454   PURDUE UNIV LAFAYETTE IND PURDUE LAB FOR APPLIED IND--ETC F/G 9/2
SIGNIFICANT ACCOMPLISHMENTS AND DOCUMENTATION OF THE INTERNATIO--ETC(U)
JAN 77   N00014-76-C-0732

UNCLASSIFIED   NL

4 OF 4
ADA036454

END

DATE
FILMED
3-77


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

Contents

1. Introduction

This paper has been produced in response to a request from the European Purdue Workshop, Technical Committee on Safety and Security, made during its meeting of 13th March 1975 in Zurich, Switzerland.

As its title suggests, the paper returns to a fundamental treatment of the terms SAFETY and SECURITY in computer-based systems. It then postulates that a structured approach to the two subject areas offers considerable advantages, both with regard to a rigorous treatment of the problems, and to a clear understanding of possible methodologies for their solution.

Although it is intended to provide a general approach to the safety and security of any computer-based system, the paper will also emphasize the real-time computing aspects, which are the concern of the Technical Committee on Safety and Security.

2. Background

Surprisingly, the existing literature on the safety and security of computer-based systems seldom contains definitions of the scope of the two subject areas and their interrelation.

As a result, a structured approach to the precise topics contained within each subject area, is poorly presented in most cases.

This is a serious omission which, apart from making much of the literature difficult to understand, has tended to result in a somewhat confused approach to problem identification and solution. We are now in a situation where the absence of a fundamental approach is hindering further progress.

To take but one example as an illustration, the term SECURITY is often misinterpreted as only relating to the provision of protection against deliberate threats, such as those posed by outsiders with some kind of malicious intent. This narrow concept treats the terms SECURITY and PHYSICAL PROTECTION as synonymous. It is a false interpretation, since PHYSICAL PROTECTION is only one aspect of SECURITY, and it leads to the danger of accidentally omitting a large part of the subject from consideration.

Accordingly, this paper will present a concept of SECURITY which is much broader in scope than mere physical protection. For example, the real possibility of accidental occurrences threatening the security of a system will also be covered.

This broader concept is not new. The paper will show that it corresponds to the common usage definition of the word SECURE.

It also corresponds to the interpretation that is used, but often only implied, in the existing literature on computer security.

Such misinterpretations as this one, abound when the subjects of SAFETY in computer-based systems are examined. This paper tries to rectify this situation.

3. Common Usage Definitions

Before examination of the particular aspect of computer-based systems, it is interesting to note that the Oxford Dictionary provides the following (extracted) common English definitions:

SAFE - Affording security, not involving danger.

SECURE - Safe against attack, impregnable; reliable, certain not to fail or give way.

From these definitions we can note that:

3.1 The definition of SAFE includes the word SECURE and vice-versa.

3.2 The definition of SAFE includes the word DANGER, which is not further defined.

3.3 The definition of SECURE includes reference to reliability and avoidance of failure as well as to attack. Since reliability is concerned with the possibility of *accidental* malfunction, an interpretation of SECURE which only accepts the possibility of attack is a false one.

3.4 In practical terms, these common usage definitions leave much to be desired. For example the use of the words "certain not to fail" may not represent a realizable proposition, since in many systems 100% certainty would not be possible.

In later sections of this paper we must take these factors into account.

4. Existing Special Definitions

As stated previously, few definitions of SAFETY and SECURITY have been offered in the literature.

A typical one, presented in the recent IBM publications called "Data Security and Data Processing",[1] was as follows:

"SECURITY: The protection of information during collection, storage, processing and dissemination from accidental or unauthorized modification, disclosure or destruction, and the protection of the system from accidental or unauthorized modification or destruction".

Although this is one of the better examples, it still leaves much to be desired. For example?

What is information?
What comprises a (computer) system?
--and so on.

In the belief that no useful purpose is servedbby attempting to derive yet another concise, precise and comprehensive definition of this kind, this paper turns to a more fundamental approach. This will be based on a very basic definition of terms which can be developed into a structured expansion, giving a clear picture of the scope of the subject areas.

Thus we will turn from an approach based on linguistics to one based on structured relationships supported by explanation.

5. Proposed Basic Definitions

As a starting point, the following very basic definitions are offered, with particular reference to computer-based systems:

SAFETY - Protection against danger to life or property.

SECURITY - Protection against attack or failure.

Deliberately, these definitions are minimal ones. At this stage they may leave some questions unanswered. That too is a deliberate attempt to begin at a simple level.

6. Safety and Security Compared

Using the basic definitions, the concepts of SAFETY and SECURITY in computer-based systems can be compared at an uncomplicated level.

Figure 1 is a diagrammatic representation of the relationship. It postulates that, in the limited field of computing, SAFETY becomes a subset of SECURITY relating to the protection of life and property.

In human terms, the protection of life means the protection of the person, either directly, by avoidance of injury or death, or indirectly by the protection of human well-being, for example by avoidance of pollution of the environment.

Quite obviously, in many real-time systems, such as nuclear reactor control, transport, process control and so on, safety is a primary consideration and may be the overriding one.

Further, as the diagram implies, other aspects of the security of computer-based systems do not relate to safety. For example, the protection of the financial viability of an organization against fraud does not affect safety in the context of our definition.

Similarly, outside the field of computing, there are many aspects of safety which are not concerned with computer security. In real-time computing there are a number of such examples which need no further elaboration here.

Using these definitions of SAFETY and SECURITY, it is the purpose of the remainder of this paper to develop a structured expansion related to a cause and effect examination.

7. A Structured Approach

The approach is intended to build from a simple beginning into areas of increasing complexity and detail.

Using the principle that SAFETY is a subset of SECURITY in computer-based systems, we will first concentrate on the structure of SECURITY and then examine the implications with regard to SAFETY.

First we will look at the causes of breakdowns in computer-based systems affecting their security. Second we will examine the effects of such breakdowns.

8. Possible Causes of Breakdown in Computer-Based Systems

This section examines the theoretical possibilities that can cause a breakdown in computer security. Later we will examine the practical considerations by relating the theoretical possibilities to some actual case histories.

8.1 Types of Threat

As defined in section 5, there are two very basic kinds of threat to computer-based systems, which can affect their security. These are:

ACCIDENTAL THREATS
DELIBERATE THREATS

- where the term THREAT is used to mean occurrences or activities which can result in unacceptable events.

Some examples of ACCIDENTAL THREATS are:

Fire
Flood
Human Error
Human Omission
Component Failure
etc.

Some examples of DELIBERATE THREATS are:

Arson
Theft
Fraud
Malicious Destruction
Dishonesty
etc.

Here, we should note that, whereas many ACCIDENTAL THREATS are not posed by human action - for example "Acts of God" - all DELIBERATE THREATS contain human involvement.

However, we should also note that many ACCIDENTAL THREATS are under human control, in the sense that they can be caused by bad design, poor manufacture, improper maintenance and so on. Even the effects of natural events, or acts of God, can often be mitigated by proper precaution. To take only one example, it would be foolish to site any computer below flood level near a river.

8.2 Unacceptable Events

The two basic kinds of threats to computer security can result in the following kinds of UNACCEPTABLE EVENTS, listed in increasing order of importance:

INTERRUPTION
DISCLOSURE
CORRUPTION
REMOVAL
DESTRUCTION

Such events can either be caused ACCIDENTALLY or DELIBERATELY, as previously indicated.

8.3 Items at Risk

In computer-based systems, the following ITEMS are at risk in terms of a possible breakdown in security:

HARDWARE
PROGRAMS
DATA/MEDIA

COMMUNICATIONS FACILITIES
ENVIRONMENT
ORGANIZATION
SUPPORT

It is important to define these basic words as follows:

HARDWARE - All equipment concerned with the computing capability, but excluding communications and environmental control equipment.

PROGRAMS - All programs required by the system, including basic software, utility programs, applications programs, test programs and so on.

DATA/MEDIA - All data entered into, stored, processed or output from the system including the media on which it is contained.

COMMUNICATIONS FACILITIES - All facilities used to transmit data, information or programs to or from the computer system, such as modems, telephone cables, radio links, remote terminals and so on.

ENVIRONMENT - All items concerned with the environmental control of the computer system, such as air conditioning, fire protection, physical access control and so on.

ORGANIZATION - The organization that is used to control the operation of the computer system. This includes the people, the responsibility structure, the standard procedures and so on.

SUPPORT - All facilities that are used to support the computer system on some form of sub-contracted basis. This could include hardware maintenance, cleaning, contracted transportation and so on.

Later in section 14 we will define these basic words in terms of more precise listings.

Now, in combination with the definitions introduced in 8.1 and 8.2 previously, these ITEMS allow the possibility of 70 basic kinds of security breakdown which we will call CAUSES. This is elaborated in Figure 2, which shows the structure of the 70 possibilities, and in Figure 3, which lists them.

At this stage these may be regarded as theoretically possible CAUSES. Later we will map some actual cases on to this structure by way of an illustration.

But first let us examine the EFFECTS that these CAUSES could have.

9. Possible Effects of Breakdown in Computer-Based Systems

Breakdowns[2] in computer-based systems can effect their:

AVAILABILITY
INTEGRITY
CONFIDENTIALITY

Let us consider these in more detail.

9.1 Availability

The use of computers has resulted in a greater concentration of processing power and data in one machine and in one place than has existed before. Potentially, hardware reliability and incidents such as fire or malicious damage are therefore much more important.

Thus, in a manual system, if a few people are away ill, work still continues, but at a reduced pace. In many computer systems, if the computer breaks down, then work stops unless and until alternative backup facilities are available, or unless the system has been specially designed with built-in redundancy.

The importance of such a breakdown in the availability of the system will depend on the application. For example, there is an increasing number of on-line systems in which the computer is relied upon to control an increased level of complexity, provide a fast response etc. Perhaps the most critical are those which control systems concerned with nuclear reactor control, transportation, missles, steel-making, chemical plants and so on. In such systems, constant availability is essential, usually because of SAFETY requirements. Hence, in such systems great attention must be given to continued availability, for example by the use of fail-safe design methods backed by alternative capability.

9.2 Integrity

It is important that any system is designed, manufactured and operated as intended and that there are enough controls to ensure that it is reasonably proof against accidental and deliberate threats, which affect its integrity.

In comparison, a manual system which has been developed over a number of years, usually incorporates many checks and controls which are not necessarily formally documented or even recognized. Thus there is a danger that, when such a system is moved on to a computer, the informal controls are not replaced by adequate formal ones. Designing a computer-based system therefore involves a level of formalization not associated with manual systems.

In a computer-based system, it is difficult, if not impossible, to give complete assurance that computer programs contain no errors or will behave as intended in all circumstances that can arise. The more complex such programs become, the more difficult it is to prove that they are correct. The very thorough testing of such systems is most important ant the employment of special programming techniques, such as structured programming, is to be encouraged.

The difficulty of testing computer systems implies that a great responsibility is vested in the technicians i.e. the system designers, constructors, installers, operators, maintainers, programmers and so on. With a manual system it is usually possible for a manager to check for himself, the integrity of the system. However he does not always possess the technical knowledge to do this for a computer system - for example by checking all of the computer programs in detail. Thus, if security is inadequate, it is possible for programmers to deliberately change the operation of the system from what was intended, and to do it in a way which is difficult to detect. There have been a number of such cases already.[3]

9.3 Confidentiality

The security of a computer-based system can be threatened if information about it is released to unauthorized persons. The items at risk have been described already in section 8.3. They show that security can be jeopardized if confidential information about hardware, programs, data, communications, environment, organization or support is accidentally or deliberately disclosed.

For example, it is certainly true that the data in computer systems is at risk. It is sometimes stored on media such as magnetic tape, which is physically small compared with paper documents carrying the same information. Therefore, it is easier to steal and, given a moderate amount of expertise and equipment, it may be easier to copy. Similarly, in real-time systems the release of information about the hardware, or other aspects of the computer system, can also jeopardize its security or its safety.

10. The Role of People

Now that we have examined an approach to computer security based on cause and effect, we can examine what these mean in both the theoretical and practical senses. These will be the subject of sections 11 and 12, where we will examine the basic possibilities and relate them to some events that have actually taken place.

But when a breakdown in security or safety does occur, it is natural to ask the question "who is responsible?", in order to complete the picture of each case.

At the most simple level breaches of security can be caused by:

NATURAL EVENTS - or "Acts of God"
EMPLOYEES
NON-EMPLOYEES - or outsiders

In the special case of NATURAL EVENTS such as flood, hurricane, lightning strike and so on, these are special cases of accidental causes of breakdown.

In the cases of EMPLOYEES and NON-EMPLOYEES these persons can responsible, either directly or indirectly, for accidental or deliberate causes of breakdown.

Thus, in the case of accidental occurrences the breakdowns in security or safety usually stem from errors *or omissions* in the design, construction, installation, operation or maintenance of the system. The minimization of such breakdown usually implies adequate control of the associated personnel, plus an adequate level of competence. We will demonstrate in section 13 that errors and omissions are by far the most important consideration in computer security.

As in the case of accidental occurrence deliberate attempts to breach security can be caused by EMPLOYEES or NON-EMPLOYEES with some kind of malicious intent. Whereas most of the cases that are reported in the press dwell on the malicious intent of outsiders, in terms of actual cases there have been relatively few. We will demonstrate in section 13 that dishonest EMPLOYEES are a greater threat.

11. Some Theoretical Possibilities for Security Breakdown

With the basic structures of cause and effect, together with an understanding of the role of people, it is now possible to examine what the practical implications could be.

In Figure 4 some of the possibilities are examined, in order to demonstrate that they can be related to what might occur.

The listing is not exhaustive.

12. Some Actual Cases

In a similar way we can examine some actual cases that have been reported by D. B. Parker in his book "Computer Abuse"[3]. This is done in Figure 5.

Again, this listing is meant to be illustrative and not exhaustive.

13. Security Facts and Figures

Some information about the number of actual cases of breaches of security is available and may be studied to advantage.

In the book "Computer Abuse"[3], 148 known cases are reported and an analysis is made.

In NCC's report "Where Next for Computer Security?"[2] the following table, resulting from a survey of some 150 organizations, is reproduced. It shows the actual nature of disruption that these organizations have experienced in order of importance.

	None	Some	Significant
Machine	15	121	16
Operator/Clerical error	11	132	15
Basic Software	24	123	12
Application Software	12	132	11
Communications	57	84	7
Power/Air Conditioning	31	118	5
Fire/Flood	129	13	1
Malicious Damage	140	2	0
Theft/Fraud/Unauthorized Use	140	2	0

It is interesting to note that disruptions associated with deliberate actions such as malicious damage, theft, fraud and unauthorized use of the system are at the bottom of the list, whereas accidental occurrences are at the top.

In IBM'x recent publications "Data Security and Data Processing"[1] the following list is published in order of importance:

ERRORS AND OMISSIONS
FIRE DAMAGE
DISHONEST EMPLOYEES
WATER DAMAGE
DELIBERATE INTRUSION

Again, accidental occurrences head the list and IBM states that these represent more than 50% of known cases.

Deliberate intrusion represents less than 5% of known cases.

These figures in total lead to the conclusion that accidental occurrences are the most important consideration by far, in any study of computer security.

14. Extending the Approach

The provision of adequate countermeasures to the 70 possible causes of a security breakdown is an exercise in Risk Management.[2] Although a full discussion of the practice of Risk Management is outside the scope of this paper it is, in brief, concerned with:

Identifying risks
Measuring risks
Countering risks

Obviously, in order to be able to identify risks we must be able to identify the things that are at risk, in some detail.

This leads to an expansion of the "items at risk" described previously in section 8.3 in order that each item of hardware, every program, every piece of data etc. can be identified for a particular system.

The basic possibilities are presented in Figure 6 as extended listings.

In a similar way, it is also possible to elaborate the special security features that are available for each of the kinds of item and so on.

The elegance of the structured approach should now be apparent. That is, when describing computer security or safety, we can do so at a very simple level, or at a level that is as complex as we desire. But it is always done in a way that can be related to other aspects without unnecessary complication.

15. Implications with Regard to Safety

Because the approach to security that has been given relates to computer systems in general, it is also intended that it should relate to real-time systems in particular.

Also, since safety is closely related to security, all of the aspects which have been isolated in developing a structured approach could be related to the safety of any computer-based system, within the definition provided in section 5.

Although many of the examples of actual breakdowns in security relate to self-standing data processing activities, such as fraud, theft, arson and so on, it is not difficult to understand that any one of the 70 basic causes of breakdown, described in section 8, could also result in a breakdown in safety.

Much work is still required to isolate the possibilities and to suggest appropriate countermeasures.

16. In Conclusion

It has been the purpose of this paper to show that a considered approach to the security and safety of computer-based systems is possible, and that problems and their solutions need not be picked at random, with the obvious danger that important considerations will be missed.

It is also important that attention should be given to the most important needs in these areas first. This should be done by an examination of case history material, together with a careful prediction of future possibilities, so that the limited amount of effort that is available can be channelled for maximum effect, if possible without unnecessary duplication.

Thus, if we decide to examine particular problems and their solution at the expense of others, we should at least know what we are discarding for the time being.

A structured approach to security and safety provides such a methodology.

Finally, the practice of Risk Management is not well-understood in computing circles. It should be.

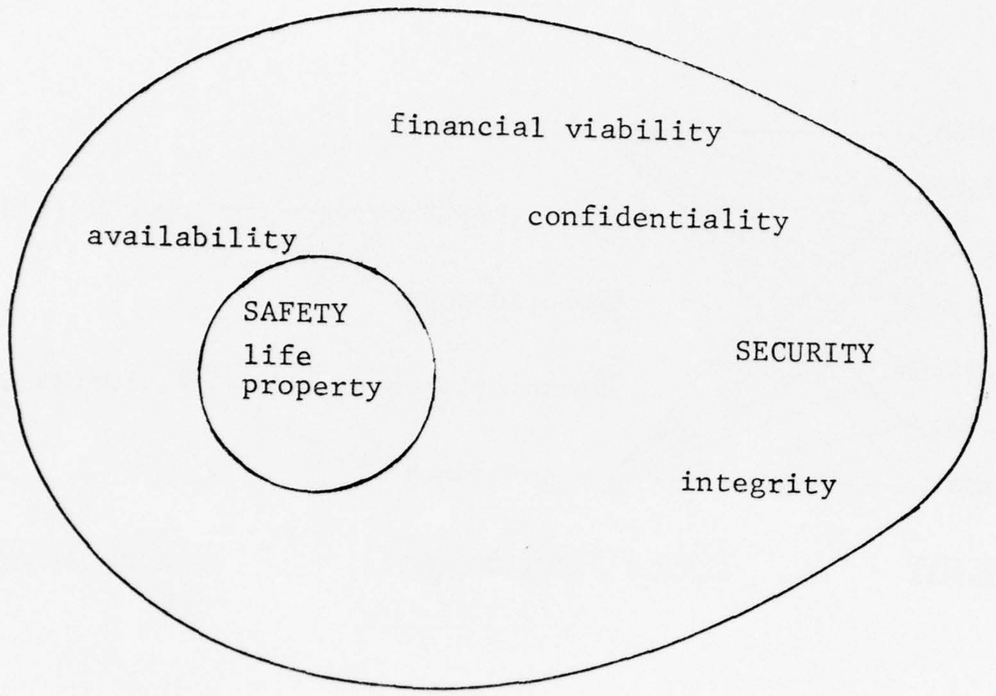


FIGURE 1

SAFETY AND SECURITY

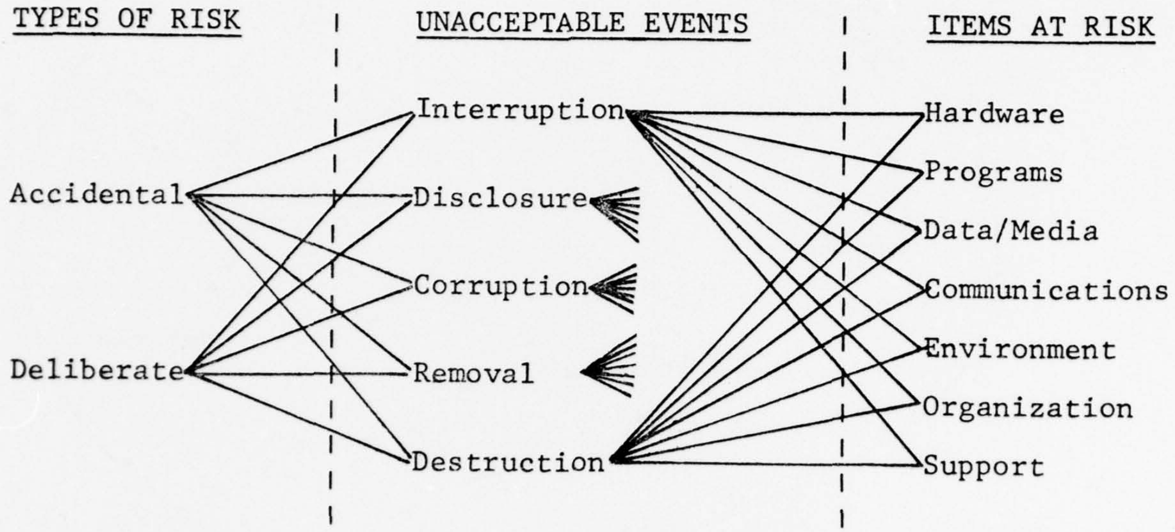


FIGURE 2

CAUSES OF BREAKDOWN

Accidental interruption of hardware
Accidental interruption of programs
Accidental interruption of data/media
Accidental interruption of communications
Accidental interruption of environment
Accidental interruption of organization
Accidental interruption of support

Accidental disclosure of hardware
Accidental disclosure of programs
Accidental disclosure of data/media
Accidental disclosure of communications
Accidental disclosure of environment
Accidental disclosure of organization
Accidental disclosure of support

Accidental corruption of hardware
Accidental corruption of programs
Accidental corruption of data/media
Accidental corruption of communications
Accidental corruption of environment
Accidental corruption of organization
Accidental corruption of support

Accidental removal of hardware
Accidental removal of programs
Accidental removal of data/media
Accidental removal of communications
Accidental removal of environment
Accidental removal of organization
Accidental removal of support

Accidental destruction of hardware
Accidental destruction of programs
Accidental destruction of data/media
Accidental destruction of communications
Accidental destruction of environment
Accidental destruction of organization
Accidental destruction of support

deliberate interruption of hardware
deliberate interruption of programs
deliberate interruption of data/media
deliberate interruption of communications
deliberate interruption of environment
deliberate interruption of organization
deliberate interruption of support

deliberate disclosure of hardware
deliberate disclosure of programs
deliberate disclosure of data/media
deliberate disclosure of communications
deliberate disclosure of environment
deliberate disclosure of organization
deliberate disclosure of support

deliberate corruption of hardware
deliberate corruption of programs
deliberate corruption of data/media
deliberate corruption of communications
deliberate corruption of environment
deliberate corruption of organization
deliberate corruption of support

deliberate removal of hardware
deliberate removal of programs
deliberate removal of data/media
deliberate removal of communications
deliberate removal of environment
deliberate removal of organization
deliberate removal of support

deliberate destruction of hardware
deliberate destruction of programs
deliberate destruction of data/media
deliberate destruction of communications
deliberate destruction of environment
deliberate destruction of organization
deliberate destruction of support

FIGURE 3 -- THE 70 BASIC CAUSES OF BREAKDOWN

Accidental Interruption	Hardware	Wrong switch thrown
	Programs	
	Data	
	Communications	telephone cable fails
	Environment	
	Organisation	
	Support	
Accidental Disclosure	Hardware	circuit diagrams released
	Programs	privileged instructions released
	Data	output given to wrong person
	Communications	
	Environment	Building layout released
	Organisation	
	Support	
Accidental Corruption	Hardware	Incorrect modification
	Programs	" "
	Data	Mispunching
	Communications	Crossed telephone lines
	Environment	Flood
	Organisation	
	Support	
Accidental Removal	Hardware	
	Programs	Program erased
	Data	
	Communications	Operator closes line
	Environment	
	Organisation	Operator in accident
	Support	
Accidental Destruction	Hardware	Computer dropped
	Programs	Fire destroys tapes
	Data	" " "
	Communications	
	Environment	Fire destroys system
	Organisation	Bankruptcy
	Support	

FIGURE 4

SOME THEORETICAL POSSIBILITIES FOR SECURITY BREAKDOWN
(ILLUSTRATIVE ONLY)

Deliberate Interuuption	Hardware	Plug removed
	Programs	
	Data	Remote sensor destroyed
	Communications	Radio signal jammed
	Environment	
	Organisation	
	Support	
Deliberate Disclosure	Hardware	Circuit diagram stolen
	Programs	
	Data	Files sold
	Communications	Telephone numbers disclosed
	Environment	Installation plan stolen
	Organisation	Security procedures given
	Support	
Deliberate Corruption	Hardware	Circuits changed
	Programs	Deliberate errors
	Data	Magnet used on tape
	Communications	
	Environment	Gas introduced
	Organisation	Bribery of staff
	Support	
Deliberate Removal	Hardware	Minicomputer stolen
	Programs	Programs removed
	Data	
	Communications	Facilities withdrawn
	Environment	
	Organisation	Strike
	Support	Maintenance withdrawn
Deliberate Destruction	Hardware	Acid, bombs
	Programs	
	Data	
	Communications	Bomb in telephone exchange
	Environment	Arson
	Organisation	Staff removed
	Support	

FIGURE 4 (Cont.)

1. TEXAS 1974

A programmer stole $5 million worth of programs he was maintaining for his employer and attempted to sell them to a customer of his employer.

An example of Deliberate Removal of Programs

2. WASHINGTON 1969

An unknown assailant fired two shots from a pistol at an IBM 1401 computer in a state unemployment office.

An example of Deliberate Destruction of Hardware.

3. TEXAS 1968

Three former employees of a Securities brokerage are alleged to have changed securities transaction statements. They claimed the changes were computer errors.

A case of Deliberate Corruption of Data.

4. MASSACHUSETTS 1969

Students took over a computer center and threatened to keep it out of operation until their demands were met by the Administration.

A case of Deliberate Interruption of the Organization.

5. SWEDEN 1970

Two employees borrowed tapes of a population registry and copied them using another computer. They sold the copies at reduced prices.

An example of Deliberate Removal of Data

6. FRANCE 1971

A programmer changed his employees program to destroy all records on a given date. This is the so-called 'timebomb in a program' case.

An example of Deliberate Corruption of a Program.

7. ENGLAND 1975

Roof of computer installation fell on to computer.

An example of Accidental Corruption of Environment.

For further examples see reference 3.

FIGURE 5

SOME ACTUAL CASES (Illustrative Only)

HARDWARE

Central processor (Main store
(Micro - code store
Add-on core store
Operator console

Magnetic tape drive (incl. cassette)
Magnetic tape drive control unit
Magnetic tape encoders
Magnetic tape readers
Magnetic tape reproducers/converters
Magnetic tape to punched card converters
Magnetic tape to paper tape converters

Magnetic disc drive
Magnetic disc drive control unit
Magnetic diskette drive
Magnetic drum drive
Magnetic drum drive control unit

Punched card punches (off line) (Print
(Non-print
Punched card punch/verifiers (buffered)
Punched card interpreters
Punched card verifiers
Punched card collators
Punched card tabulating equipment
Punched card punches (on-line)
Punched card readers
Punched card reader/punches
Punched card reproducers/converters
Punched card to magnetic tape converters
Punched card to paper tape converters
Punched card sorters

Punched paper tape punches (off-line)
Punched paper tape punch/verifiers (off-line)
Punched paper tape punches (on-line)
Punched paper tape reader/punches
Punched paper tape readers
Punched paper tape reproducers/converters
Punched paper tape to magnetic tape converters
Punched paper tape to punched card converters
Punched paper tape splicers
Punched paper tape printer
Punched paper tape hand punches

FIGURE 6

ITEMS AT RISK

Optical character readers
Optical mark readers
Magnetic ink character readers
Bar code readers

Cheque readers
Document readers
Marked card readers
Page readers
Tag readers
Tally roll readers
Magnetic stripe card readers
P O S equipment
Shop floor data collection equipment
Data logging equipment
Audio response units
Digitisers

Accounting machines with byproduct paper tape
Accounting machines with byproduct mag tape
Cash registers with byproduct mag. tape (POS?)
Automatic typewriters with p.t. output

Remote batch terminals
Keyboard printer terminals
Line printers
Page printers
Serial printers
Computer output to micro film
Visual display units
Light pens
Digital displays
Touch wire displays/graphic/tabular
Digital input units
Digital output units
Digital contact scanners
Analogue contact scanners
Output typewriters
Digital/analogue conversion
Graph plotters

Data concentrators
Facsimile transmission equipment
Modems
Acoustic couplers

Magnetic tape transmission equipment
Punched card transmission equipment
Punched paper tape transmission equipment

FIGURE 6 (Cont.)

Communications processors
Communications transmission lines
Communications controllers
Front end processors
Network controllers
Remote communications controllers
Radio/microwave transmitters/rec. Aerial systems
Punched paper tape winders

Bursting/decollators
Guillotines
Shredders
Computer furniture

Storate racks/cabinets, cards
p.t.
mag. tape
mag. discs

Transmit containers: cards
p.t.
mag. tape
discs

P.t. dispensers
P.t. Winders
Mag. tape winders
Desks, tables, lockers
Trolleys, waste bins

FIGURE 6 (Cont.)

PROGRAMS

Operating Systems:

- Disc
- Tape
- Multi-access
- Process control
- Real time

etc.

Micro-Code

Compilers

Simulators

Translators

Diagnostics

Trace programs

Macros

Program generations

Applications programs, object and source

Plotting programs

Audit routines

Linkage editors/Consolidators

Pre-processors/pre-compilers

Executive programs

FIGURE 6 (Cont.)

DATA/MEDIA

Master files

Transaction files and documents

Report Files

Tables

Print outs of files or software

Software documentation

Operating procedure documentation

Documentation of personnel duties

Input data

Results/output data

Punched Cards

Paper tape

Magnetic disc packs including diskettes

Magnetic tape including cassettes

Continuous stationery

Documents

Microfilm

Magnetically striped cards

Edge punched cards

FIGURE 6 (Cont.)

COMMUNICATIONS

Telephone lines

Telegraph lines

Radio/Microwave Links

Postal services

Freight services

Private data carrying services

Messengers

Private transmission lines

Satellites

Exchanges

Message switching centres

Repeater stations

FIGURE 6 (Cont.)

ENVIRONMENT

Building structure, fittings and equipment, carpets, etc.

Building layout

Building siting in relation to other buildings, etc.

Building siting in relationship to natural surroundings

Electrical supplies

Fuel supplies: Coal/Oil/Gas

Water supplies - hygiene & kitchen
- sanitation
- air conditioning

Sanitation and waste disposal

Heating and Ventilating plant

Air conditioning plant

Catering facilities

Cleaning services

Fire detection equipment (incl. smoke)

Fire fighting equipment - sprinklers, extinguishers, sand buckets, hoses

Lift services - passenger and goods

Equipment lifting facilities

General removal and installation facilities

Drainage system

Rainwater system

FIGURE 6 (Cont.)

ORGANIZATION

Management structure

Management policy

Data processing management

Computer operations management

Systems analyst management

Programming management

Hardware maintenance management

Systems analysts

Programmers - systems and applications

Operators

Data media librarians

Data preparation staff

Data preparation clerks

Office services management

Office services staff - typing
- mail
- clerical
- reception

Personnel management - Selection
- Training
- Interviews
- Remuneration
- Appraisal
- Discipline
- Career development
- Conditions of employment
- Absenteeism
- Resignation
- Job Satisfaction

Users of the System

Janitorial services

General maintenance

FIGURE 6 (Cont.)

References

1. "Data Security and Data Processing" - IBM Publications 1974 - G320 - 1370 to G320 - 1376 inclusive.

2. "Where next for Computer Security?" - NCC Publication 1974.

3. "Computer Abuse" - D. B. Parker - Stanford Research Institute Report 1973.

EUROPEAN PURDUE WORKSHOP

TC Safety and Security

Author: Dr. H. Trauboth

Institution:

Institut für Datenverarbeitung in der Technik der Gesellschaft für Kernforschung Karlsruhe

Date (Assigned): April 1975

Date (Completed):

TC SS	Nr. 35
Category:	T
Updates:	No. 8
Replaces:	No. 8
pp:	No. 1-15

Title: Methods to Develop Safe Computer Systems

Contents:

1. Introduction

The development of a complex real-time computer system is a multi-phase process which involves many people who have to work as a well-organized team in a systematic way. This process should be controlled by project management methods at each phase to ensure that the product of each development phase meets the system requirements. The success of such project management methods has been demonstrated in large aerospace and weapon system developments such as in NASA's APOLLO program and the US-Navy's POLARIS program. These methods are now being applied also in the commercial world, e.g. for the development of large computer hardware and software systems for commercial and industrial applications (1,2). Although these methods have been developed for large and complex systems, the philosophy of project management can also be applied to smaller system developments.

For systems which are subject to a high degree of safety requirements, the development must follow sound design and project management rules as a prerequisite for producing a reliable and safe system. It is assumed here that a necessary condition for a system to be classified as "safe" is the correctness of its performance according to its requirements. In addition, the special safety aspects should be considered for each phase of the development process.

The objectives of the safety measures in the development of computer systems are to ensure that:

a) Safety is designed into the system.

b) Hazards associated with each system, subsystem and equipment are identified, evaluated and eliminated or controlled to an acceptable level.

c) Control over hazards that cannot be eliminated is established to protect personnel, equipment and property.

d) An effective system safety program is planned and integrated into all phases of system development, production and operation (3). (Similar objectives can also be found in the specification Mil-Std-883 "System Safety Program for Systems and Associated Sub-Systems", 1969 of the US-Air Force).

The safety measures employed in the system design should ensure that any error in computer hardware and/or software does not cause harmful or unpredictable actions. Of course, the criticality of errors and the safety measures to protect against these errors depend on the particular application of the computer system. It is assumed that errors in the system can be generated at all phases of the development process and during operation and maintenance.

The paper will outline the phases of the development process (fig. 1) and refer briefly to possible safety measures within each phase. It will also address test and project management methods. The description of the development phases is kept to a minimum since to a large degree that informaiton can be found in other literature (1,2). It is beyond the scope of this paper to present more than brief hints at the safety aspects of each development phase. The paper should serve as an overview or frame which has to be filled with more detail to be worked out by further committee task assignments.

In each phase, the consideration of the safety aspects must include an answer to the three questions:

Does the design

- minimize the occurrence of errors (error prevention)
- detect all critical errors
- take proper actions in case of a critical error, i.e. does the action lead to a safe state of the system (safe error recovery)

The criticality of an error is determined by the consequences of an error (grades of criticality).

Each safety measure has a "price tag" attached to it. Thus, in selecting a safety measure out of several alternatives, one has to consider its cost and effort to implement it as criteria besides its effectiveness.

2. Development Process

2.1 System Requirements Analysis

2.1.1 Brief Description

The system requirements analysis establishes what the computer system (hardware, software) is supposed to do, i.e. its objectives, and under which conditions with respect to its environment, costs, reliability and safety it is to operate.

The plant or process to be monitored and/or controlled and the characteristics of its equipment are described as well as the operational requirements and performance requirements of the major computer systems functions.

It is important that the system requirements are complete and to sufficient detail to serve as input for the design phases.

2.1.2 Safety Aspects

During this phase, safety and reliability requirements should be determined, i.e.

- critical failure modes of the process equipment
- safety devices and safety configurations in the plant (e.g. interlocking controls, redundant parts, back-up components)

- operational measures (safety procedures to be observed by operations personnel)
- minimum M T B F
- maintainability and testability of hardware and software components
- critical process variables which have to be monitored by the computer for safety reasons
- critical process states that can lead to unsafe conditions.

2.2 Functional Systems Design

2.2.1 Brief Description

The functional or conceptual systems design should define how the system should be structured with its major computer hardware and software functions to meet the system requirements. These functions and subfunctions include processing control, acquisition, storage, processing and transmission of data and man-machine communications. The functions may be assigned to hardware equipment and/or software such as programs and data files. The algorithms of the data processing function are also described. If required, the command language for the man-machine function is defined and the data flow through the system is determined.

Thereafter, the functional design is evaluated to estimate roughly performance and capacity of the various components and to identify bottlenecks in the data and control flow. Out of alternative design configurations a final design configuration is selected.

An implementation and test plan is to be established at this phase. The implementation plan depicts the major tasks of the system development process and the course time schedule of these tasks. The test plan indicates all tests to be performed during the development process until the installation of the system, its time schedule and the tools such as simulators to be required for these tests. If necessary special test tools may have to be developed which are also included in the test plan.

2.2.2 Safety Aspects

To some degree, special safety aspects can already be considered at this early time of the development process. Certain critical functions may have to be built in several ways by different methods such as for navigation and guidance in a spacecraft and advanced aircraft. These different methods allow a different mode of operation in case of an error without leading to disaster. For instance, the system can be switched from automatic landing mode to manual mode. Critical components such a storage devices or complete computers may be designed in redundancy or as back-up. Those functions that are not essential for guaranteeing safety such as the protocol printing and history filing of a computerized protection system for a nuclear reactor should be taken out of the safety area where a malfunction does not cause any hazard. Moreover, critical functions like interlocks for actuators may be duplicated by simple fixed wired electronic circuits in parallel to more sophisticated software interlocks. The run-time testing features of the computer hardware functions are defined.

2.3 Computer Hardware System Design

2.3.1 Brief Description

The functions assigned to hardware and the necessary hardware to support the software functions are now translated into specifications of the computer hardware system. They include the configuration of the system and the characteristics of the system components such as processors, memories, peripherals, transmission devices and their interfaces.

2.3.2 Safety Aspects

Hardware errors and safeguards against them such as redundancy and back-up of critical hardware components are considered in this phase. Comparator circuits, detection and signalling of hardware errors, switching circuits for redundancy anc back-up, coordinator circuits etc. are designed. Detection and correction of transmission errors by error codes, parity checks and repetition of transmission are included. Fore more detail the reader is referred to the paper on "Safe Hardware" of this committee.

2.4 Functional Software Design

2.4.1 Brief Description

The functional or systems software design defines how the software should work to meet the system requirements. This definition includes the

- o structure of the applications software
    - o program modules
    - o control strategy
    - o data flow
- o major data files
- o algorithms
- o command language
- o systems and support software

2.4.2 Safety Aspects

In this phase, the safety aspects to be considered refer to error prevention, detection and recovery.

A few safety measures are presented as examples for recommendation.

For error prevention:

- The program modules should be defined along functional lines so that they perform rather independent functions and their coupling (via program parameters and data files) are kept to a minimum. They should be the first level of top-down software design.
- The main control function activates the various tasks at fixed time slots and for fixed length of time which are allocated to each task (fast cycles for frequently recurring tasks and slow cycles for less frequently recurring tasks).
- The tasks may be initiated by polling rather than by interrupt (also for asynchronous initiation).
- Functions and subfunctions should use their own files as much as possible rather than common files. Major functions may be assigned to separate hardware devices resulting in hardwire decoupling", e.g. data acquisition and processing.

For error detection:

- Provide checks for proper time allocation and length of operation of tasks.
- Provide special software functions for hardware error detection such as background test programs and redundant programs with different algorithms for the same computation including comparison functions.

For error recovery:

- Repeat a faulty operation (e.g. in transmission or arithmetic operation to recover from sporadic errors).
- In redundant operations, switch off the operation which was determined faulty by majority voting and continue operation with reduced reduncancy.

- Switch in back-up device (or file). The systems and support software must be checked for their safe operation, e.g. the compiler must be checked that it, generates correct code for all input conditions.

2.5 Detail Software Design

The safety aspects in software design is treated by the working group "Safe Software" (Dr. Ehrenberger).

3. Testing and Verification

3.1 General

TESTING is the activity to detect, locate and "fix" errors in the object being tested. VERIFICATION is the activity to prove and demonstrate that the object being verified performs according to the requirement specifications of the design.

Testing and verification should take place during all development phases starting with the functional design, so that errors are caught early.

The testing activities should detect the following types of errors:

o Requirements errors	RE	
o Systems errors	SE	
o Hardware errors	HE	
o Design errors	HDE	
o Implementation errors	HPE	
o Operational errors	HOE	o (physical wear)
o Software errors	SE	
o Design errors	SDE	
o Program errors (bugs)	SPE	
o Interface errors	IE	
o Documentation errors	DE	

Testing and verification should take place during the whole development process to cath errors early. After a phase has been completed, its results should be checked and compared with the specifications at the input of that phase. Especially the safety measures should be scrutinized.

If possible, the checkout should be performed by personnel separate from design personnel in order to guarantee integrity of testing (without bias), i.e. by checkout personnel. Testing and verification is a tedious and costly process. Therefore, it should be planned well during the design and implementation phase. The plan should specify the

- Test environment needed
- Test software and hardware to be used
- Test data generation (cause and effect analysis)
- Test strategy (submodule, module, system)
- Test output data reduction, interpretation and documentation.

In the testing process, we distinguish between various types of testing:

- Hardware testing
  - Device tests
  - Interface tests
  - Hardware system test
- Software testing
  - Static testing (prior to execution)
  - Dynamic testing (during execution)
- System test

3.2 Hardware Testing

3.2.1 General Remarks

The description of and guidelines for hardware testing should be performed by the working group on "Safe

Hardware". For completeness, only a very brief outline of hardware testing is presented.

3.2.2 Device Test

The various devices of the computer hardware system (processors, memories, peripherals, modems, etc.) are checked out separately by applying simulated stimuli signals to their input lines. Special test equipment for check-out of digital devices may be used.

3.2.3 Interface Tests

Before interconnecting the individual devices to a complete system, the consistency of the hardware interfaces (control signals, data lines) are checked.

3.2.4 Hardware System Test

The complete system is driven with test data that activate all devices and their cooperation. This test is first performed without connecting the computer to the process to be monitored or controlled by using a simulator which generates the signals coming from the process. After successful system testing, one portion of the process at a time is connected up until the whole process is on-line.

It is a matter of testing philosophy, whether the testing should start with the device testing followed by the interface and system test (bottom-up testing) or immediately with the system test (top-down testing). In the latter case, the isolation of a greater number of faults is more difficult, however, one saves the effort and time of the device and interface tests.

3.3 Software Testing

3.3.1 Static Testing

First, the software modules and then the integrated software system are tested manually or automatically by special analyzers for compliance with the software design rules.

Typical test cases are defined from the system and software requirements to generate "comparison" data.

The consistency between the results of various development phases (levels of refinement) and between the specifications at the input of a phase and the resulting design or code is checked at each phase.

The functional design and detail design are tested, e.g.

- o timing estimates
- o generation of proper output data
- o logical sequence
- o computational steps to satisfy algorithm specifications
- o data in files, tables and lists
- o safety measures by introducing errors are checked. The program code is "desk-checked" using the comparison data.

Analytical methods of "proof of correctness" are still in the stage of early development and seem not yet feasible to be applied to large software systems.

3.3.2 Dynamic Testing

The dynamic testing of large software systems is performed in several steps. First, all submodules are tested separately, then the individual modules and finally the total integrated system resulting in the acceptance test (bottom-up test).

We distinguish between different levels of detail testing with different tools, i.e. simulators:

Level 0: Interpretive simulation of computer program under test (e.g. flight computer program) on large host computer, which also simulates the process being monitored and/or controlled by computer system.

Level 1: (See Fig. 3) Process is simulated by mathematical model on digital computer.

Level 2: Process is simulated by mathematical model on hybrid or analog computer (to test fast process responses).

Level 3: Process includes critical hardware components together with simulated process parts.

Level 4: Process includes as much hardware components as possible for final system test.

Variations of these four levels or a reduction to two levels are possible.

Various test data for different tests have to be generated, i.e. for

- Verification
    - test data generated from performance requirements specifications,
    - typical (nominal) benchmark data,
    - critical (non-nominal)benchmark data,
- Malfunction analysis
    - Data which represent critical malfunctions (hardware and software)
- Completeness test
    - Test data activating all possible paths and data transactions of the software to check for correct and complete performance of software.

- o Statistical test
  - o Statistically generated test pattern of input data (Monte Carlo)

4. Project Management

The objective of project management is to ensure the quality and safety of the developed system according to the system's performance requirements and safety requirements within given overall costs. Project management performs the following control functions:

- o Control of the development process (reviews)
- o Control of changes (configuration and change control)
- o Quality control (test and verification process)
- o Cost control

4.1 Control of Development Process

Reviews are held at the end of each phase by a review board of the development organization and at critical points also by the customer (Fig. 4). The approval to proceed with the next phase depends on the results of the review of the previous phase. We distinguish between various reviews:

o Systems Requirements Review o Functional Systems Design Review (Preliminary Systems Design Review)	Hardware and Software
o Software Requirements Review (SRR) o Functional Design Review (FDR) (Preliminary Design Review) o Detail Design Review (DDR) (Critical Design Review o Final Software Review (FSR) (Software Delivery Review	Software

The review should monitor the progress and check whether the design meets the original objectives and performance requirements. They should detect problems early and initiate immediate steps for their remedy. Each development phase must be well documented (see documentation guidelines). The reviews are based on documentation and verbal discussion. The review board consists of the system project manager and technical project management staff each of whom is assigned to a particular technical area of concern. In large projects which require high safety standards, special personnel is assigned to control the safety of the system during all phases of the development process. This personnel checks independently if all required safety measures have been designed properly into the system.

As examples, the tasks of the software reviews are outlined.

The Software Requirements Review (SRR) reviews all system requirements that are necessary to define the scope of work for the software development of the baseline design. It checks for completeness of the catalogue of requirements.

The Functional Design Review (FDR) (Preliminary Design Review) checks the baseline functional design if it satisfies all requirements. It checks the modification of existing programs to be utilized, the parameters of algorithms and equations used (e.g. for different flight mission) and the command language used for operating the system.

The Detail Design Review (DDR) (Critical Design Review) is a technical review of the detail design to ensure that the design in in agreement with the functional design specifications and system requirements. The results of the verification and tests of the design phase are also discussed.

In the Final Software Review (FSR), the final programs and their agreement with the detail design and original requirements are reviewed.

The results of final software testing and verification are discussed. The final software products including final documentation (delivery items) are reviewed:

- Listings
- Program descriptions
- Verification (simulation) output data (print-outs, plots)
- Tapes and card decks
- Documentation of changes (patches) incl. various configuration reports
- Users manual

4.2 Quality Control (Testing and Verification Process) (Fig.6)

An analysis of the software requirements, software baseline specifications, past changes of software and modifications of verification simulators is performed during the design and implementation phases in preparation of the tests and verification according to the Program Verification Plan.

Various tests are prepared for verification:

- tests of nominal functions of software
- tests of non-nominal functions of software (extreme situations in process which is computer-controlled)
- tests of malfunction cases in process and computer hardware
- tests of back-up programs for failure cases.

In the final analysis, it must be checked, that all requirements and operational conditions have been simulated for the verification of the software.

The purpose of the Program Verification Plan (PVP) is the establishment and documentation of all simulations, definition of tests and their relationship to the execution of software functions. It also shows the allocation of tests to various simulators (see Fig. 3).

The Software Problem Report(SPR) documents errors detected during the verification and is used for initiation of the change control process.

The Program Verification Document (PVD) documents results of the verification effort and changes of simulations due program changes. It lists simulation data output and conditions (nominal, non-nominal, failures) and depicts the relationship of tests to specific program releases.

4.3 Control of Changes (Configuration and Change Control)

4.3.1 General

Changes are a way of life in the development of a large computerized system especially in a research environment. Proper control of changes is very important to guarantee quality and safety of the computer system hardware and software to be developed. Changes that impact safety measures are particularly marked and treated by the CCB.

Change is defined as the deviation from a baseline, i.e. from the

- Requirements baseline
- Functional baseline
- Detail design baseline
- Product baseline

We distinguish between different types of changes:

type 1 - affect hardware interface specifications, cost and/or schedule, and/or baseline

type 2 - no effect on cost nor schedule, changes as results of verification, but have global effect

type 3 - no effect on cost nor schedule, changes have only local effect

type 4 - customer directed changes

Type 1 and 2 require approval by the Change Control Board (CCB). The control of hardware changes follows similar procedures as those for software.

4.3.2 Functions of Change Control Board (CCB) (Fig. 5)

The Change Control Board consists of representatives from three major software development areas (contractor, developing organization)

- design
- implementation (programming)
- verification

The CCB is responsible for controlling the assessment, impact and release of software changes: It

- coordinates software activities related to changes
- coordinates change impact and assessment
- prepares Engineering Change Proposal for all approved changes
- initiates any hardware changes
- releases all approved software changes for implementation
- establishes program delivery dates with customer
- maintains adequate documentation and control for tracking of program changes (bookkeeping)
- acts as technical contact with customer for program changes.

4.3.3 Software Change Requests (Reports)

Requests for changes are initiated and documented by

- o Software Problem Report (SPR), which
  - is initiated in case of errors or deficiencies
  - reports any design, implementation or documentation errors or deficiencies and their effects subsequent to document approval. (Functional, detail, detail design).
- o Design Change Request (DCR), which
  - is initiated in case of new or expanded requirements
  - requests the change of a baseline
  - documents requirement changes, their justification, programs being affected, program changes and verification procedure.
- o Preliminary Engineering Change Proposals (PECP) which
  - is used to prepare, process, and incorporate type 1 changes which require customer's approval.
  - documents the changes proposed, the programs affected and the cost and time estimates to implement the change.

The implementation of approved changes is documented in the Software Maintenance Report (SMR) which

- documents changes made to any baseline
- describes the source, environment and application of all change data closes change.
- closes change.

REFERENCES

1. Metzger, P. W., Managing a Programming Project, Prentice Hall, 1973

2. Hice, G. F., et al, System Development Methodology, 1974

3. Rodgers, W. P., Introduction to System Safety Engineering, John Wiley, 1971

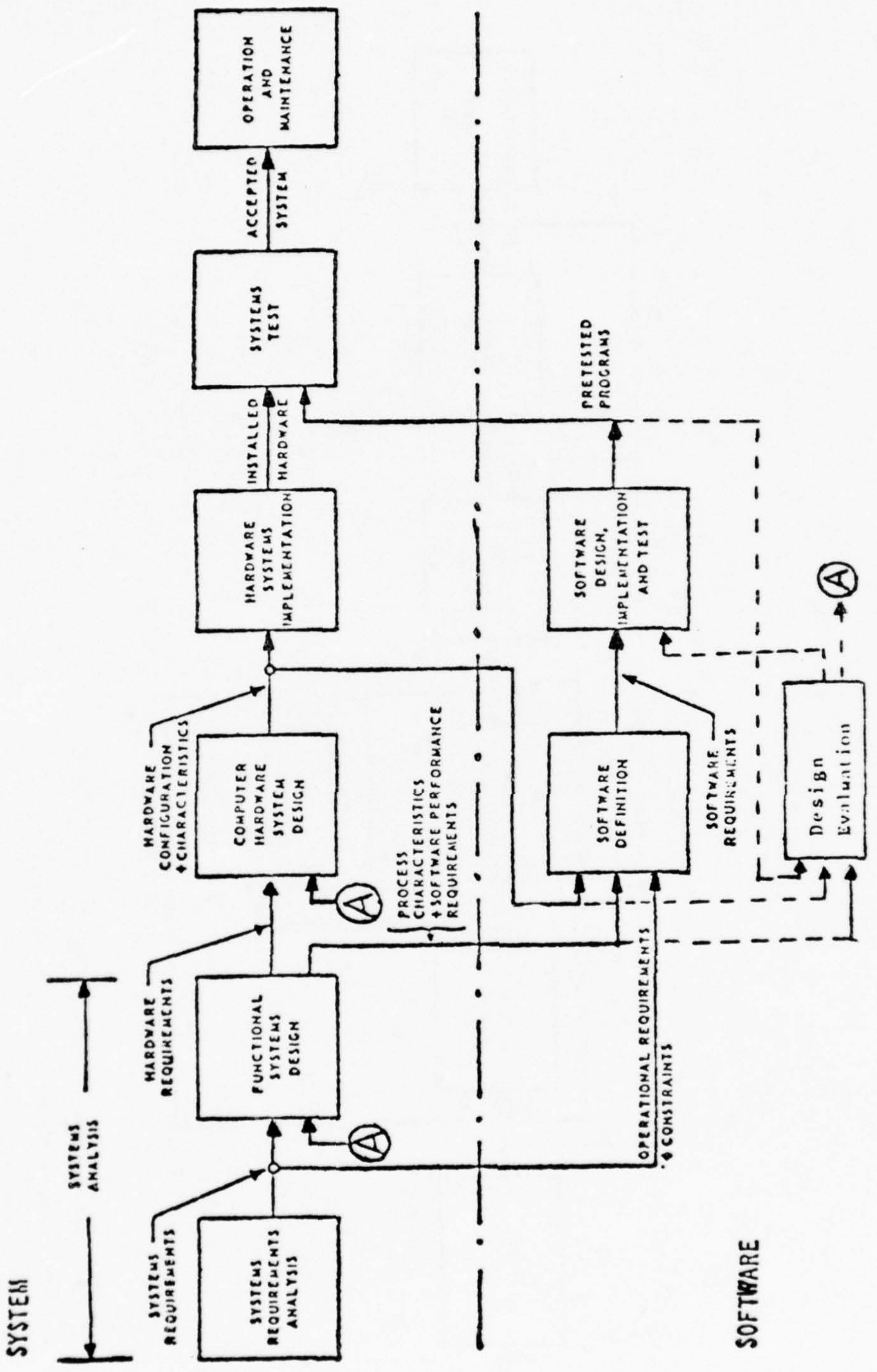

SYSTEM
SYSTEMS ANALYSIS
SYSTEMS REQUIREMENTS ANALYSIS
SYSTEMS REQUIREMENTS
FUNCTIONAL SYSTEMS DESIGN
A
HARDWARE REQUIREMENTS
COMPUTER HARDWARE SYSTEM DESIGN
A
HARDWARE CONFIGURATION + CHARACTERISTICS
HARDWARE SYSTEMS IMPLEMENTATION
INSTALLED HARDWARE
SYSTEMS TEST
ACCEPTED SYSTEM
OPERATION AND MAINTENANCE
PROCESS CHARACTERISTICS + SOFTWARE PERFORMANCE REQUIREMENTS
OPERATIONAL REQUIREMENTS & CONSTRAINTS
SOFTWARE
SOFTWARE DEFINITION
SOFTWARE REQUIREMENTS
SOFTWARE DESIGN, IMPLEMENTATION AND TEST
PRETESTED PROGRAMS
Design Evaluation
A

FIGURE 1

PHASES OF SYSTEMS DEVELOPMENT

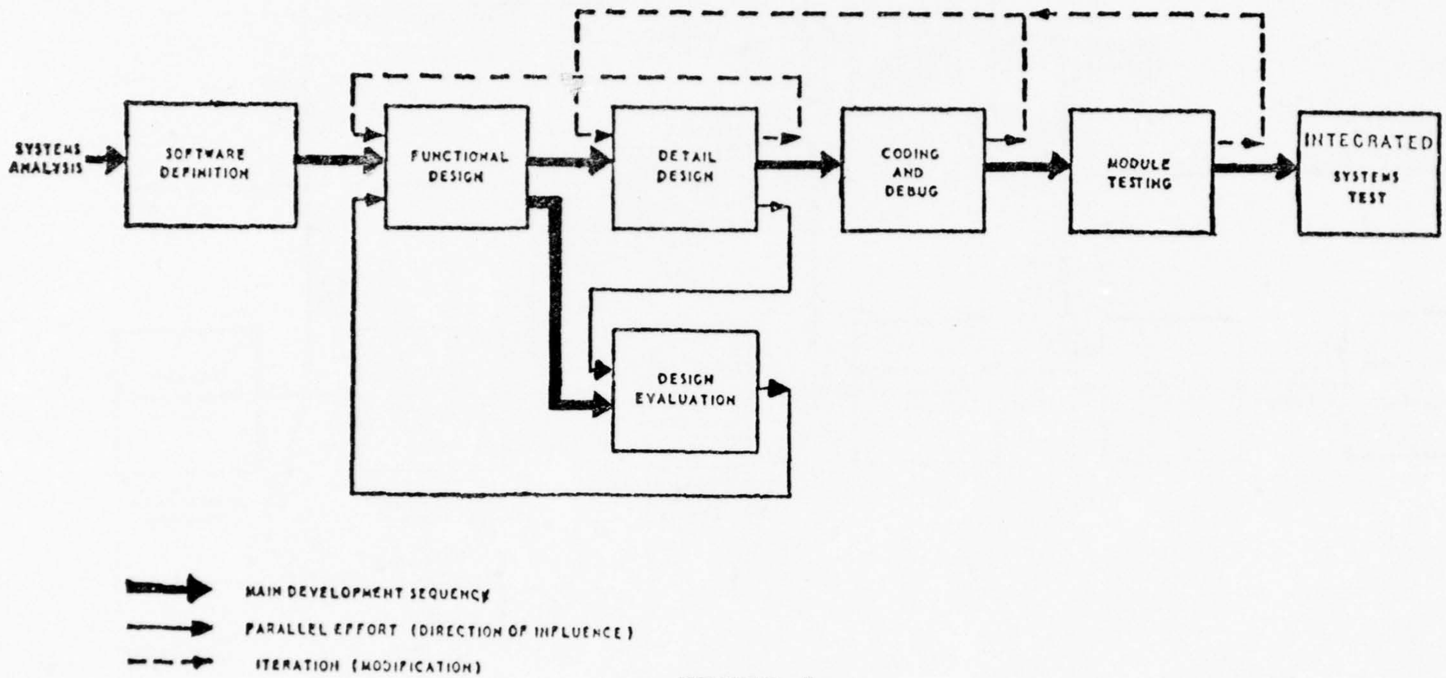


FIGURE 2

PHASES OF SOFTWARE DEVELOPMENT PROCESS

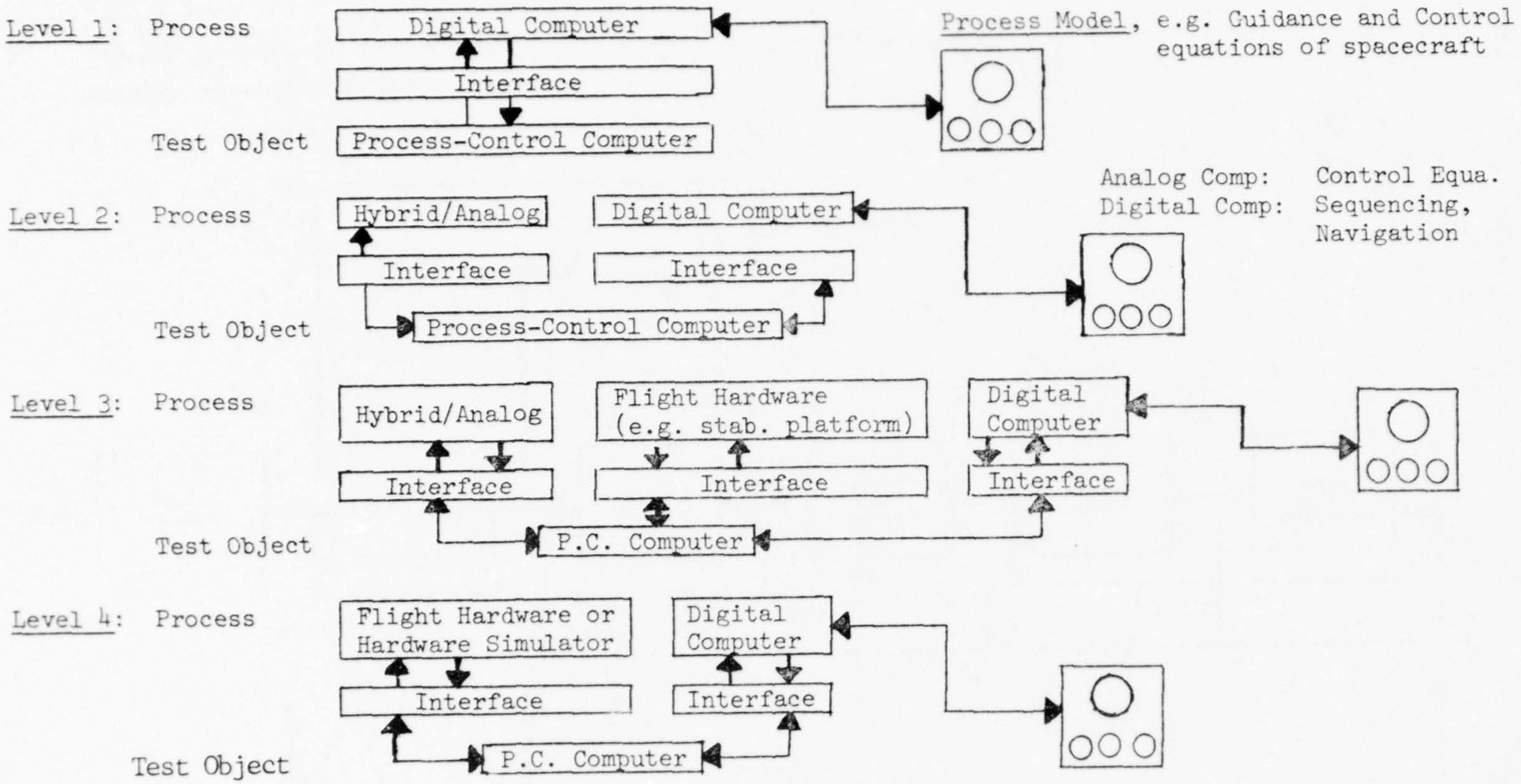


FIGURE 3

DIFFERENT LEVELS OF DYNAMIC TESTING
(ES. FLIGHT COMPUTER OF SATURN V-INSTRUMENT UNIT)

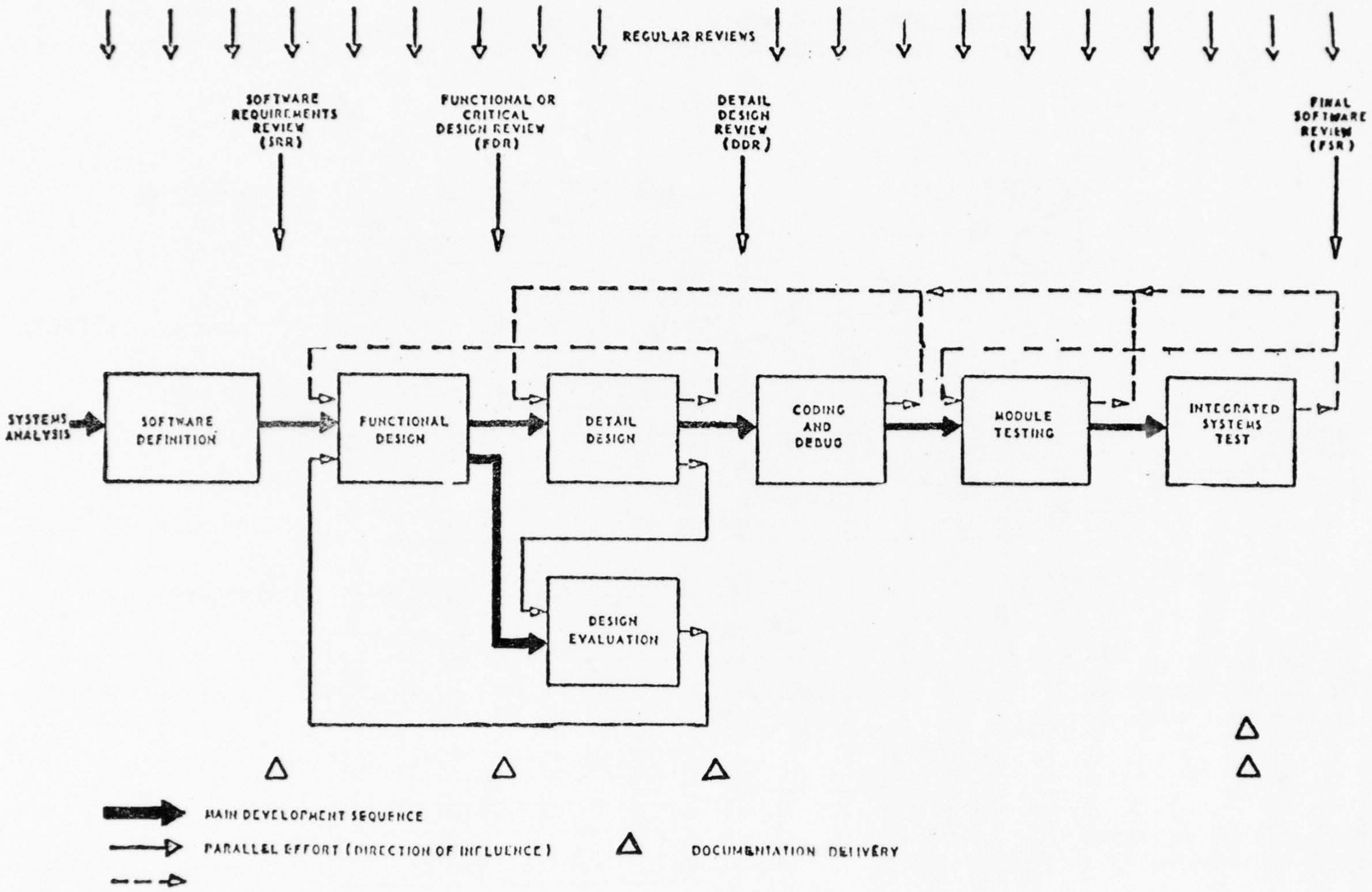

REGULAR REVIEWS
SOFTWARE REQUIREMENTS REVIEW (SRR)
FUNCTIONAL OR CRITICAL DESIGN REVIEW (FDR)
DETAIL DESIGN REVIEW (DDR)
FINAL SOFTWARE REVIEW (FSR)
SYSTEMS ANALYSIS
SOFTWARE DEFINITION
FUNCTIONAL DESIGN
DETAIL DESIGN
CODING AND DEBUG
MODULE TESTING
INTEGRATED SYSTEMS TEST
DESIGN EVALUATION
MAIN DEVELOPMENT SEQUENCE
PARALLEL EFFORT (DIRECTION OF INFLUENCE)
DOCUMENTATION DELIVERY

FIGURE 4

PHASES OF SOFTWARE DEVELOPMENT PROCESSES INCL. POINTS OF REVIEW

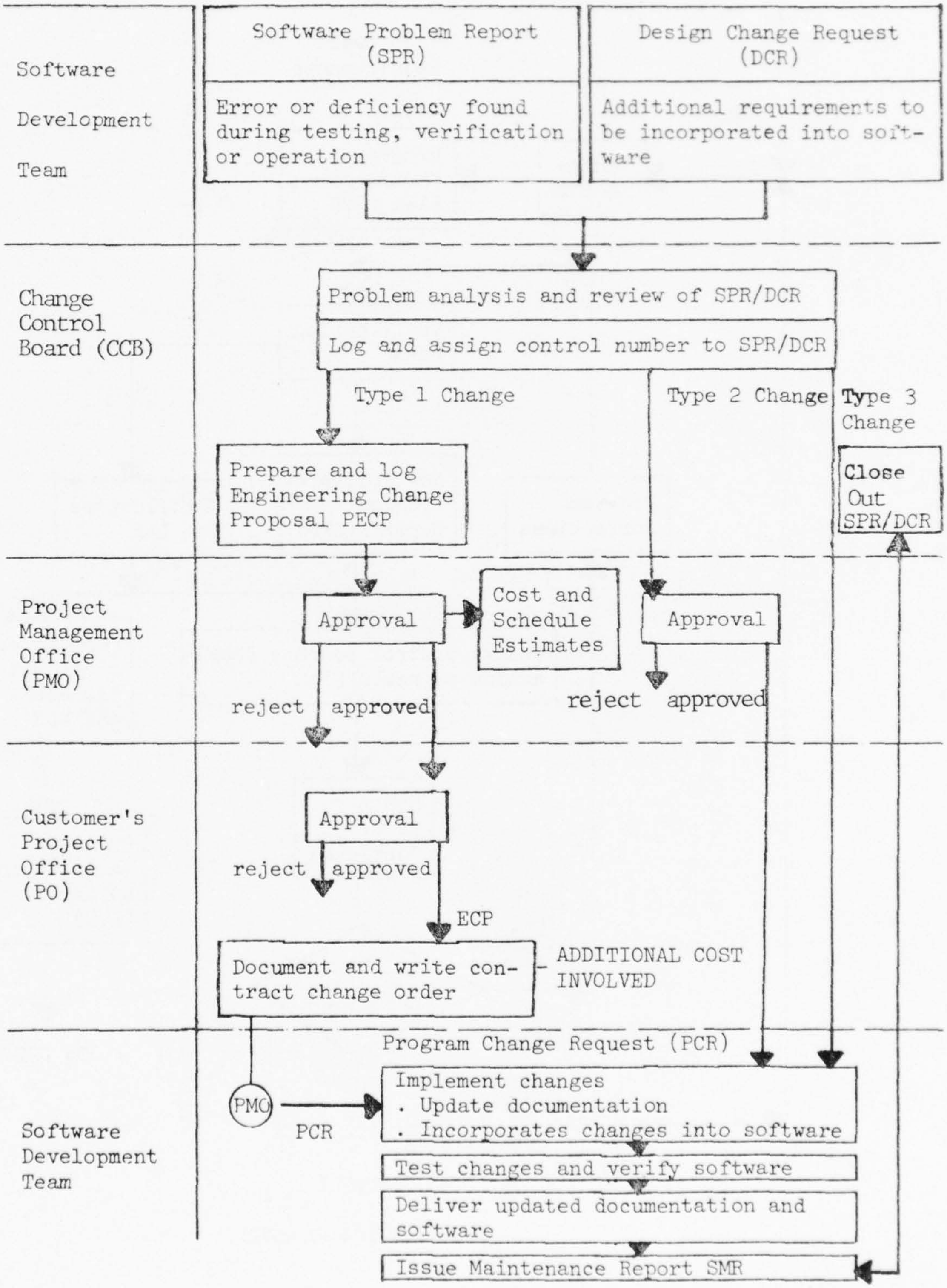

Software
Development
Team
Software Problem Report (SPR)
Error or deficiency found during testing, verification or operation
Design Change Request (DCR)
Additional requirements to be incorporated into software
Change Control Board (CCB)
Problem analysis and review of SPR/DCR
Log and assign control number to SPR/DCR
Type 1 Change
Type 2 Change
Type 3 Change
Prepare and log Engineering Change Proposal PECP
Close Out SPR/DCR
Project Management Office (PMO)
Approval
Cost and Schedule Estimates
Approval
reject
approved
reject
approved
Customer's Project Office (PO)
Approval
reject
approved
ECP
Document and write contract change order
ADDITIONAL COST INVOLVED
Software Development Team
Program Change Request (PCR)
PMO
PCR
Implement changes
. Update documentation
. Incorporates changes into software
Test changes and verify software
Deliver updated documentation and software
Issue Maintenance Report SMR

FIGURE 5

CHANGE CONTROL SYSTEM

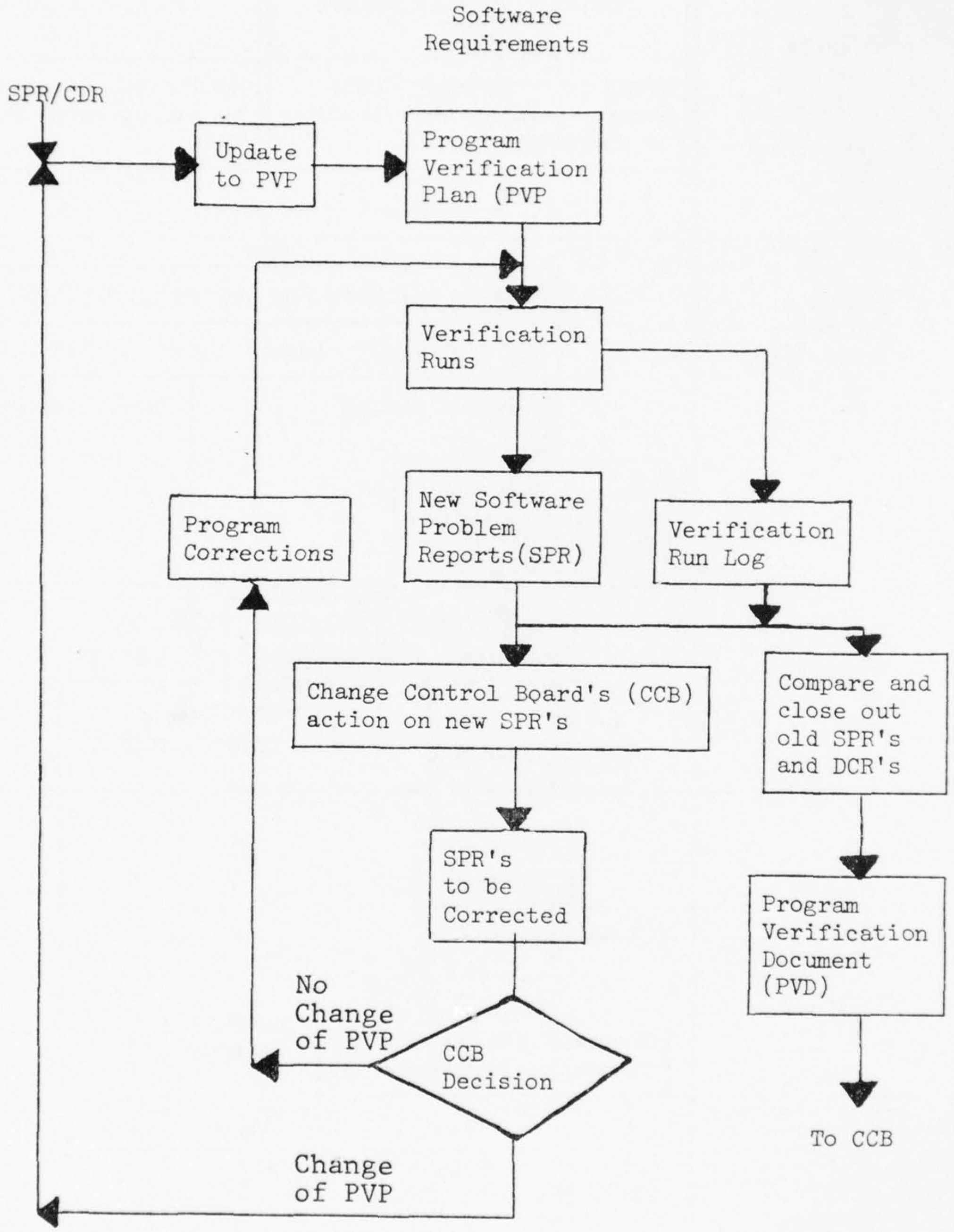


FIGURE 6

VERIFICATION PROCESS

EUROPEAN PURDUE WORKSHOP
TC Safety and Security

Author: R. Lauber

Institution:

Institut für Regelungstechnik und Prozessautomatisierung der Universitat Stuttgart

Date (assigned):

Date (completed): 7.10.1975

TC SS No. 37

Category: T

Updates:

Replaces:

pp: 9

Title: Safe Software by Functional Diversity

Contents:

1. Introduction

Safe computer hardware may be realized by using 2 out of 3 computer systems with fail-safe voters. But still, the basic problem of the safety of computer software has not been overcome (see fig. 1). Many proposals have been made to attack this problem (1,2,3,4,5). Nevertheless, the full formal demonstration of software safety to an assessment authority presents major difficulties. Functional diversity programming is suggested here as a general solution.

2. Safe Software Versus Error-Free Software

As was pointed out by Wobig (6), one has to distinguish two problems when designing software for safe computer systems:

- the design of error-free programs (absence of "prenatal program errors")
- the protection against faulty programs which once had been shown to be error-free, but which were falsified by hardware errors ("post-natal program errors")

By using 2 out of 3 computer systems, hardware faults as well as post-natal software faults may be prevented from causing danger (assuming that identical hardware faults will not occur in 2 out of the 3 computers in a certain time interval). Thus, for a 2 out of 3 computer system one solution to the problem of software safety would be the design of error-free software (proof of the absence of "prenatal program errors").

This method corresponds to the "direct" method introduced by Konakovsky(7). Another solution to the problem allows both prenatal or postnatal software errors to be present, but prevents these errors to cause danger by using some form of redundant programming ("indirect" methods according to the terminology in (7).

3. Attempts to solve the problem of software safety

3.1 Direct methods (see fig. 2)

Ehrenberger (2) proposed to develop safety related software according to special recommended principles in order to prevent software errors. Certainly, the number of software errors may be drastically reduced when these recommendations are strictly applied. But there is no way to prove that programs written according to these recommendations do not contain any errors.

Another unpublished proposal suggests extensive tests to detect prenatal errors. As was shown theoretically by Dijkstra ("tests can only prove the presence of errors, but they never prove the absence of errors"), also a practical case reported in (8) shows that this method certainly does not solve the problem.

The most direct way seems to be the use of program verification methods (suggested in several papers by Hoare, Taylor, Ehrenberger, etc.). Unfortunately, these methods turn out to be only applicable to relatively small programs and under certain restrictive conditions (9). Therefore, a statement in (10) says that it seems to be practically impossible to verify the correctness of programming systems of "normal" size (where "normal" size means realistic user programs of 16 to 64 K of instruction words). If this statement holds, the verification methods do not solve the safety problem of presently developed software systems. These methods may eventually give a solution in the future.

In this respect, statement No. 6 in (10) ("A software system of "normal" size is never static in the sense that no changes wil occur") is of importance. Even if the verification methods are further developed to be - hopefully - applicable to realistic software systems, this application must be cheap enough to be possible every time a change in the software system occurs.

3.2 Indirect methods (see fig. 3)

"Redundant" programming was recommended in some papers (5), especially the multiple design of complete user program sby different and independent programmers to be run in separate computers.

The difficulties of this approach are evident: multiple design of software by independent programmers seems to be hardly realizable.

4. The functional diversity method

This method is illustrated in fig. 4: It essentially uses redundancy like the already mentioned method of "redundant" programming. It thus belongs to the class of indirect methods, but differing from the above-mentioned redundant programming method by the fact that

- a "diversified redundancy" is used (consisting of a function which is completely independent from the "normal" functions of the user programs)
- no complete redundancy is used

The "diversified function" consists

- of a plausibility check, if the output is an analog signal or a digitally coded value.
- of a checking procedure if the output is a binary signal. This checking procedure uses a strategy different from the strategy of the "normal" user programs.

The main principle of this method may best be illustrated by considering the well-known plausibility checks: The plausibility of the results of an algorithm may be checked by rough estimates about the physically possible values of the output variables. Thus, the functions performed by the normal user programs are "protected" by estimation functions.

There are several methods and strategies to construct diversified estimation functions[1]. They offer the following advantages:

- The "normal functions as well as the "diversified" estimate functions may be programmed by the same programmers (the probability of programming errors in both programs compensating each other is considered to be negligible).
- The diversified estimate functions may be realized by relatively simple and short programs (thus the total cost of memory and the time required is not doubled by the redundant programming).
- A formal demonstration of software safety to an assessment authority presents no difficulties (it consists of proofs that all safety related outputs are checked by diversified function results). Moreover, the method of functional diversity may be explained easily to non-experts, using analogies from other fields (for example: two independent and different brakes in an automobile).

[1] A more detailed explanation of these methods including examples will be published in the near future.

Literature:

(1) Delpy A. and Schwier W.: Probleme des Sicherheitsnachweises beim Ensatz von Prozessrechnern in der Eisenbahnsignaltechnik, Signal und Draht 63 (1971) S. 175 - 183

(2) Ehrenberger W.: Design of safety related user programs. Working Paper TC SS No. 19 (April 1975)

(3) Welbourne, D.: Computers for Reactor Safety Systems. Nucl. Eng. Int. November 1974 pp. 945 - 950

(4) Schallopp P.: *Protection System Developments* and Trends in the Federal Republic of Germany. Nuclear Safety 15 (1974) pp. 409 - 417

(5) Frech G.: Zuverlassigkeit und Sicherheit in Systemen mit hoher Sicherheitsverantwortung. Signal und Draht 66 (1974) S. 40-47

(6) Wobig K.-H.: Safe Software Versus Error-free Software. Working Paper TC SS No. 31 (July 25, 1975)

(7) Konakovsky R.: A New Method to Investigate the Safety of Control Systems. IFAC Congress, Boston 1975. Preprints Part III D

(8) Boehm, B.W.: Software and Its Impact: a quantitative assessment. Datamation, May 1973

(9) Bunsen A.: Verifikation eines Betriebssystems fur einen Mikrorechner. Diplomarbeit, Inst. f. Regelungstechnik und Prozessautomatisierung, Univ. Stuttgart (July 1975)

(10) Lauber R.: Working Paper TC SS No. 21 (June 2, 1975)

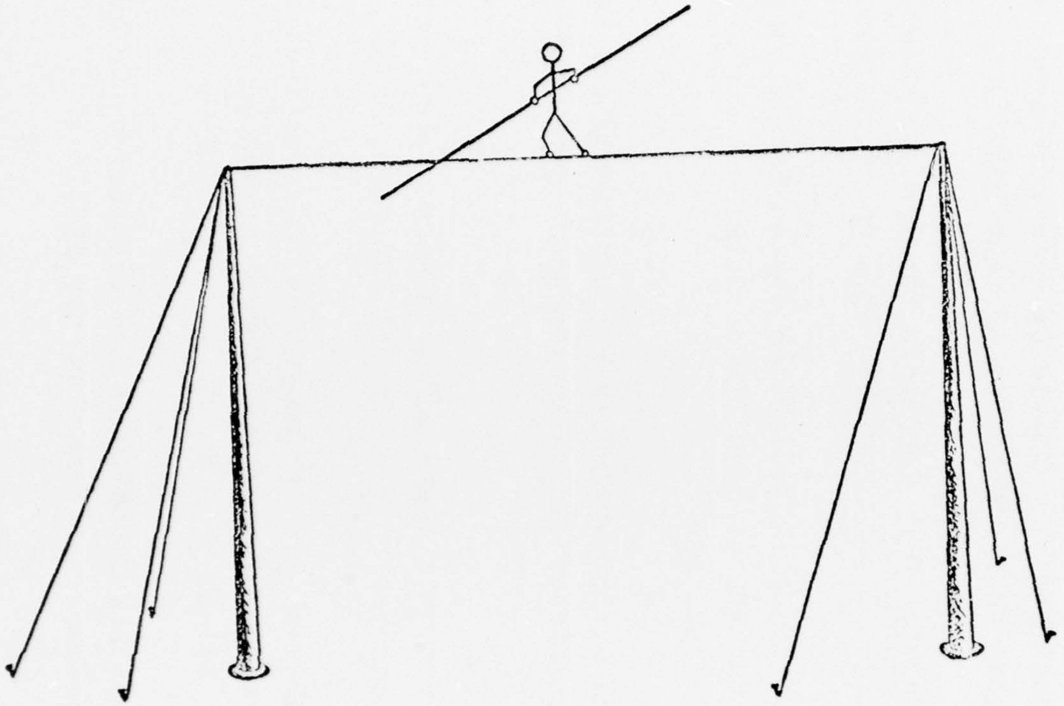


FIGURE 1

ILLUSTRATION OF THE PROBLEMS OF SAFE SOFTWARE

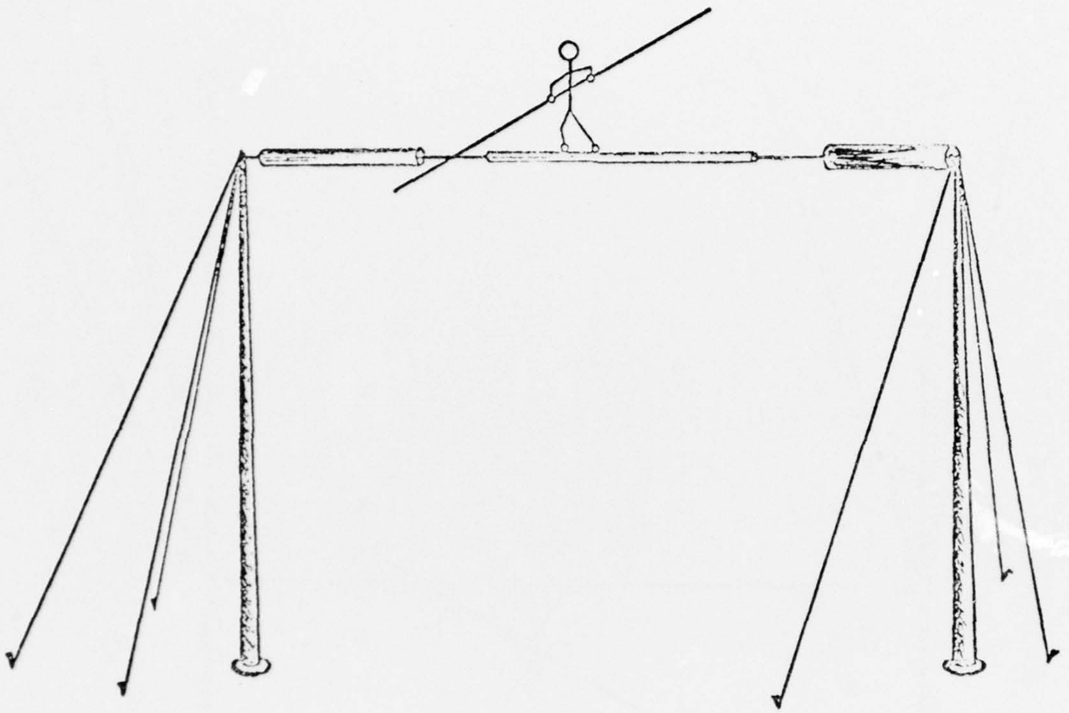


FIGURE 2

SAFE SOFTWARE BY ATTEMPTS TO VERIFY PROGRAM CORRECTNESS

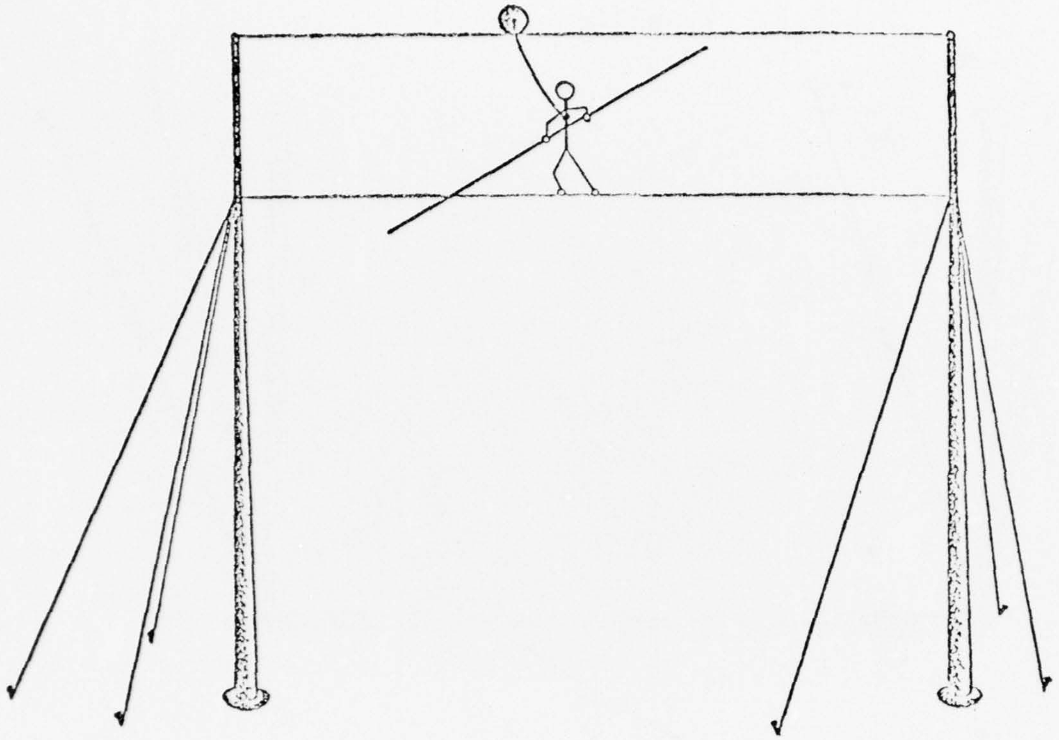


FIGURE 3

REDUNDANT PROGRAMMING USING TWO COMPUTERS WITH DIFFERENT PROGRAMS WRITTEN BY INDEPENDENT PROGRAMMERS

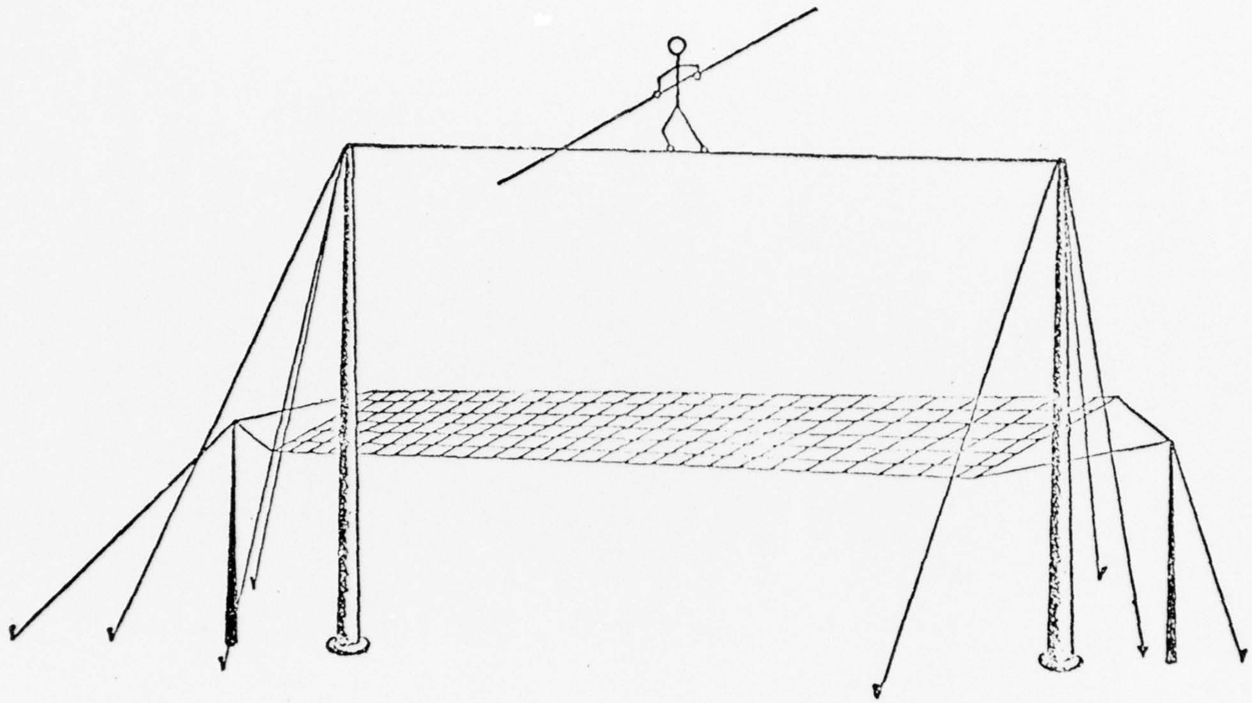


FIGURE 4

SAFE SOFTWARE BY FUNCTIONAL DIVERSITY PROGRAMMING

INTERNATIONAL PURDUE WORKSHOP ON
INDUSTRIAL COMPUTER SYSTEMS

PURDUE LABORATORY FOR
APPLIED INDUSTRIAL CONTROL
102 Michael Golden
Purdue University
West Lafayette, Indiana 47907, USA
317/494-8425

Please reply to:

THE GUIDELINE FOR SAFETY OF THE INDUSTRIAL COMPUTER SYSTEMS

The contents of the report issued by the guideline working group members under the Sub-committee for Safety and Security of the Industrial Computer Systems Committee of the JEIDA were summarily desclibed as follows.

1. The Scope on the Safety

(1) Figure-1 shows the situation of the industrial computer system to be considered from the viewpoint of safety.

(2) The utilization of the industrial computer systems is anticipated to be one of the important factors for the improvement of safety in the process plant.

2. Background of the Issued Report

(1) Problems concerning safety have been thoroughly discussed, surveyed and systematically settled by the working group members.

(2) The following two major items were referred as the

Affiliations
Purdue University
Instrument Society of America through Data Handling and Computations, Chemical and Petroleum Industries, and Automatic Control Divisions
International Federation for Information Processing as Working Group, WG 5-4, Common and/or Standardized Hardware and Software
Techniques of Technical Committee, TC-5, Computer Applications in Technology

themes on safety.

(i) The safety of the industrial computer system itself.

(ii) Functions for the safety-ensuring of the plant.

(3) However, functions for the safety-ensuring would depend on the applied process, so that it would be hard to be surveyed and settled readily.

(4) Therefore the working group had the activities to survey, investigate and settle the problems concerning the safety on the industrial computer system, which were represented as follows;

(i) High reliability system

(ii) Maintainability

(iii) Safety assessment

(5) Collection of data from many users and makers were referred for the working group's activities.

3. High Reliability System

(1) This item will be broken down as Hardware, Software and System Configuration. And for Table-1 shows the subjects taken up for each sub-items.

(2) The followings should be further investigated.

(i) Definition of measure on the both hardware and software.

(ii) Structural programming technique for error free programming.

(iii) Distributed system architecture.

(iv) Relationship between man and machine.

(v) Fail safe system design.

(vi) Communication data security.

4. Improvement of Maintainability

(1) For this item, environmental conditions, training and maintenance, preventive and corrective are mentioned as subjects as shown Table-2.

(2) The followings should be noted.

(i) Definition on system life.

(ii) On-line system maintenance.

(iii) Measurements for down-time reduction.

5. Safety Assessment

(1) Cost-safety measurements and methods for safety assessment analysis are discussed as subjects as shown in table-3.

(2) The followings should be noted.

(i) Cost-safety analysis as the optimum investment problem.

(ii) Utilization of FTA and FMEA methods.

Fault Tree Anayl
Failure Mode

Figure-1 Computer Control System

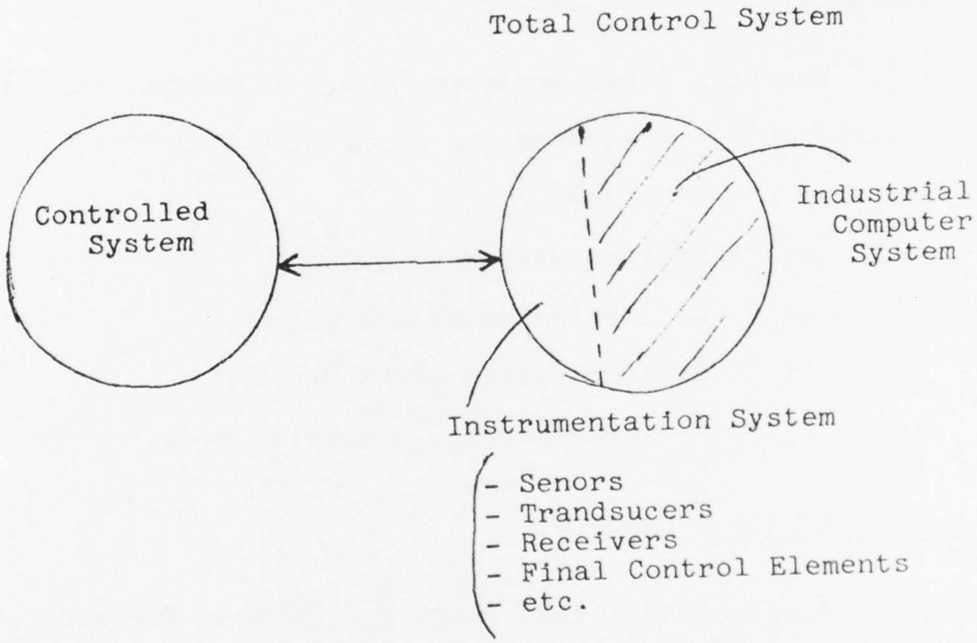


Table-1 High Reliability System

Major Subjects	Concrete Subjects
Hardware	° Measure on reliability ° Checked subjects for high reliability ensuring ° Function for emergency detection
Software	° Measure on reliability ° Software design ° Programming method ° Program test method
System configuration	° System form ° Back-up system ° Fail safe design ° Data communication ° Man-machine communication ° Data file security

Table-2 Improvement of Maintainability

Major Subjects	Concrete Subjects
Environment	° Installation and environmental conditions ° Working conditions
Maintenance	° Maintenance form ° Maintenance condition ° Maintenance employee
Training	° Training form ° Training tool

Table-3 Safety Assessment

Major Subjects	Concrete Subjects
Cost safety	° Factors and cost for the safety-ensuring ° Present status and trends in the investment to the computer system
Design Assessment	° Methods for analysis ° Raws and regulations ° Planning example for the safety-ensuring